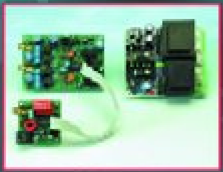
MARCH 2001
£2.95

ELEKTOR ELECTRONICS

THE ELECTRONICS & COMPUTER MAGAZINE

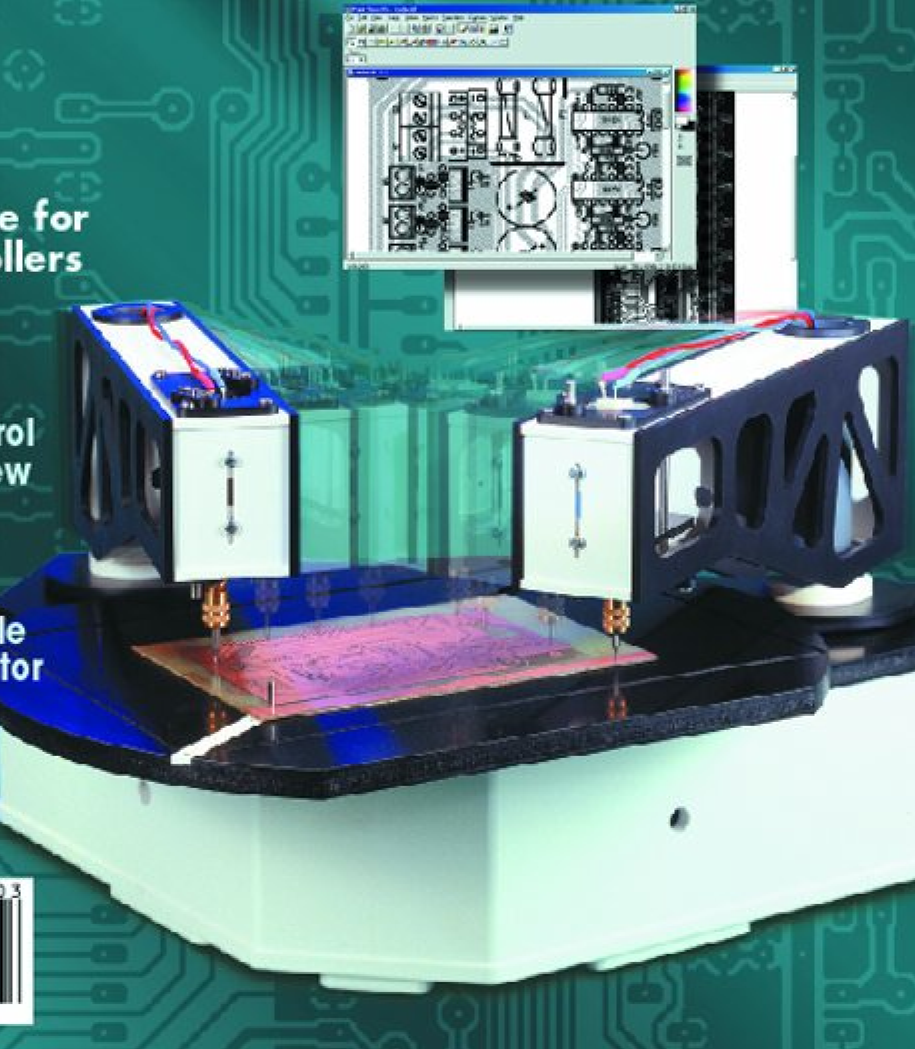
www.elektor-electronics.co.uk



Audio ADC 2001

IDE Interface for Microcontrollers

PCB Drilling Machine



Remote Control Code Overview

Running Text Display

Programmable Pulse Generator

MIDI Lights & Slide Control

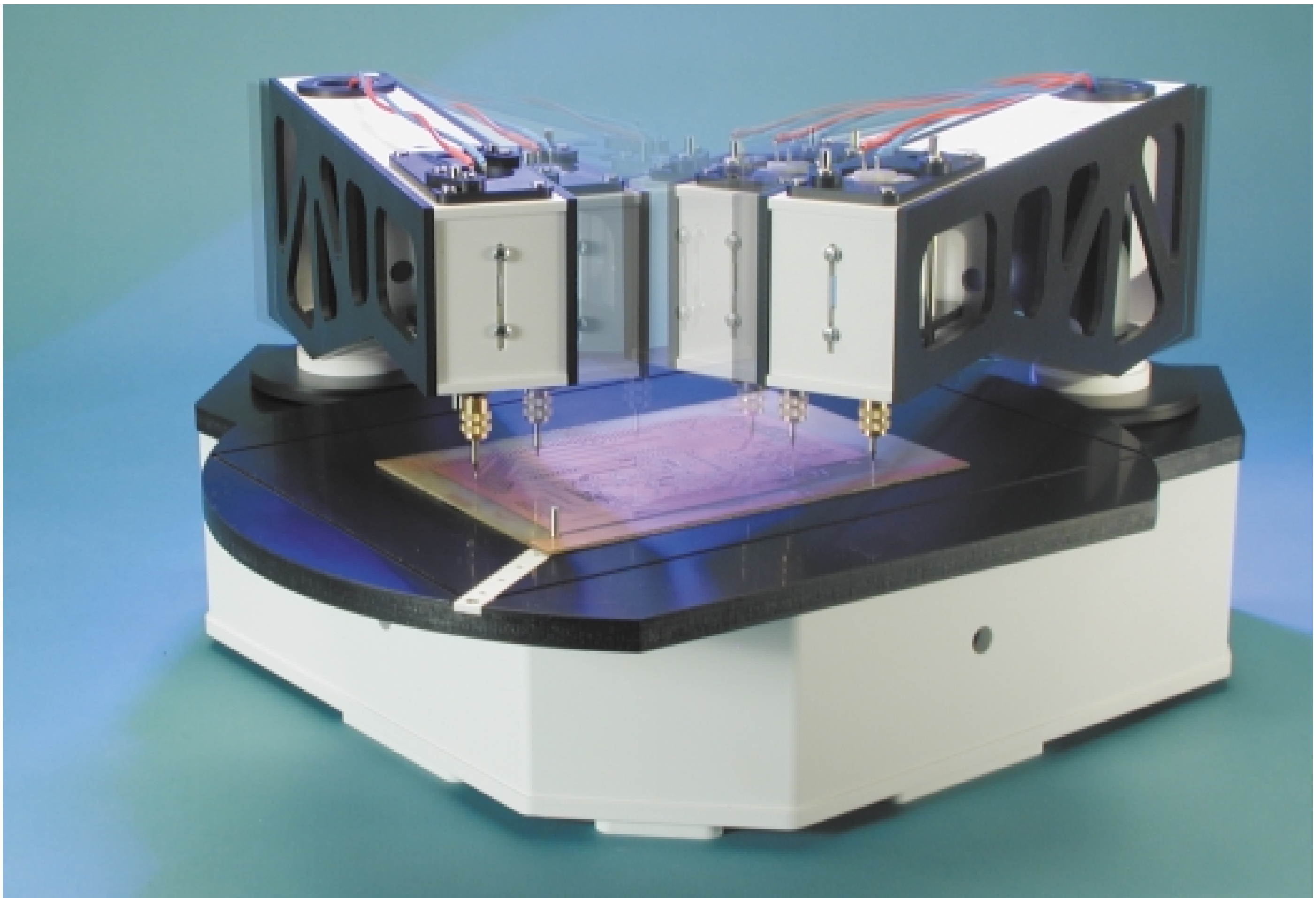
03
9 770268 431081

PCB Drilling Machine

Part 1: New solutions to old problems

By T. Müller (Radix GmbH)

Elektor Electronics is proud to present an unprecedented home construction project: a CNC PCB drilling machine which is economical, accurate, quick, and bristling with new ideas.



Large-scale electromechanical projects have tended to be the exception rather than the rule in *Elektor Electronics*; but these exceptions have always been well received. Many readers will have fond memories of the Plotter and Plotter Mark 2 projects, which hundreds of constructors across Europe have built and which even fifteen years after publication still boasts a large fan club.

There is a very good reason behind the idea of presenting an electromechanical project for home construction. Rather than the fascinating and in places supremely elegant and unusual design, it is the cost saving that drags the ambitious hobbyist away from the soldering iron to less familiar tools. Home construction of a plotter made a lot of sense when a ready-made commercial device was outside the financial grasp of the electronics hobbyist.

These days, however, plotters only find specialist application, and low-cost ink-jet printers do the 'plotting' work; and no-one would consider producing a printer for home construction when mass-produced commercial devices are so cheap.

So such a project in *Elektor Electronics* must be timely and have a 'best before date' well in the future. The computer-controlled PCB Drilling Machine described in this article and the instalments to follow fits the bill. Commercially available machines with comparable specifications are around four times as expensive as the one presented here, and are hardly likely to become mass-produced commodities sold at bargain basement prices. In conventional CNC drilling machines the mechanical parts are too expensive and too complex, and cannot be made from cheap materials.

This opens the way for new ideas. The author has developed a CNC PCB drilling machine that is distinctive in two ways: first, in its economical mechanical construction, using moulded plastic parts rather than the conventional multitude of aluminium and steel components; and second because of the supremely elegant and unusual design.

We would like to describe a little of the story of the PCB drilling machine in a series of constructional articles. If you decide to construct the machine, you do not of course have to make the components yourself. Through his company, the author is offering to supply a complete kit of mechanical parts; *Elektor Electronics* (and certain component distributors) the electronics. The basic construction kit should not cost more than about five hundred pounds. More on this, and delivery details, will appear on the second instalment in the next issue of *Elektor Electronics*.

Specialist mechanical skills are not required. You will be as surprised by how foolproof construction of the machine is as we were in the *Elektor Electronics* laboratory when we first saw it run. And even if you are not planning to build the drilling machine, we hope you will still find the articles interesting!

PCB Drilling machine

Concept

CNC machine with 2.5D operation capable of vertical drilling in circuit boards up to 300x200 mm^2

Construction features

- No torsion on machine, since all points where forces are applied lie at centres of rotation
- No expensive sliding components such as V-grooves, linear bearings or ball races
- Self-calibrating drill advance using magnetic drive
- Low torque about base fixing points

Functional features

- Up to four tools in use simultaneously
- No tool changing required
- Greatly improved speed moving from drilling point to drilling point
- Clear and easy access to table from above

Benefits

- Circular turntable allows fixing points outside the usual rectangular workpiece: conventional tables lose working area to fixing points
- Small footprint due to efficient use of area
- No ball races, no worm gears
- No thermal problems due to non-planarity
- Simple construction, no calibration for planarity or squareness
- No force on drilling head from trailing cables
- All heavy and sensitive components underneath or clear of working area
- Swarf cannot jam mechanism

From Idea to Circuit Board

The individual steps in developing an electronic circuit, from the idea through to the populated prototype circuit board, usually follow on from one another more or less smoothly.

First of all there is the initial design. Depending on the experience of the designer and on the complexity of the circuit, small parts of the design may first be tried out on prototyping board or by tacking components together. Then the parts are placed on a printed circuit board, leaving space for expansion, power supply, and the like.

In the past printed circuit board layout was a tedious job, involving the rubbing down of transfers onto transparent film. These days things are much easier: using a suitable software package, the tracks and pins are simply sketched on the computer and the layout can subsequently be modified and processed as desired. The layout program will allow the results to be output in many different ways: frequently the layout is simply printed out on paper, but for higher quality (and at greater expense) the results can be photoplotted onto high-resolution film at a reprographics bureau.

The light-sensitive surface of a suitable piece of copper-clad board can now be exposed and developed. This removes the light-sensitive resist coating in those areas where the copper is to be etched away. At this point, for most electronics hobbyists and prototype builders, the nightmare begins. To develop and remove the unwanted copper chemical processes are used which employ various unpleasant chemicals. We surely do not need to explain to Elektor Electronics readers that the properties of the etchant are such that not only the copper on the circuit board, but also hands, tables, chairs and clothing — all can get eaten away. Some advice: if you use a colourless etchant, you only notice a splash when it is too late, and you find a hole in your clothing; use a coloured etchant and you can see the impending disaster before your clothes are ruined. A brown iron (III) chloride (ferric chloride) stain is easy to spot and can be neutralised with sodium hydroxide (caustic soda), before using a specialist stain remover.

Etching need no longer be carried out, as it was in the past, in open trays: closed etchant-spray tanks are readily available, and

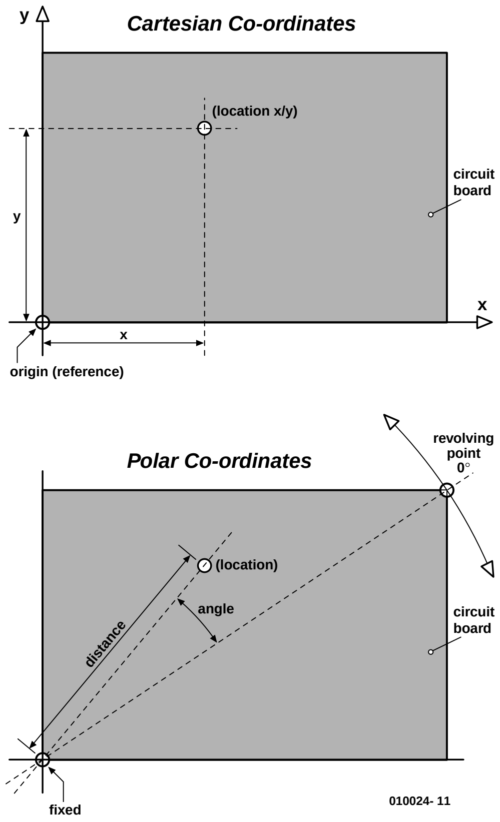


Figure 1. Cartesian and polar coordinates.

affordable. Closed systems keep the unpleasant chemicals to a defined area.

Do not forget that any remaining etchant must be washed from the newly-etched board first with water and then any still clinging to the board can be treated with developer.

Caustic soda is cheap and in any case does not keep well. Plain water will not do the job: it does not clean the board properly. After drying, the copper side should be sprayed with a lacquer to make soldering easier and to prevent the copper from oxidising.

A Boring Story

Next, holes are bored in the circuit board using drills of various diameters. In general, three drills are used: the commonest is 0.7 mm or 0.8 mm, used for almost all small components such as resistors, capacitors, ICs; it is even suitable for the leads of some electrolytic capacitors. A 0.9 mm or 1 mm drill is used for most connectors, square pins and the larger diodes. Larger connectors, solder pins and similar components require a 1.2 mm or 1.5 mm drill.

Usually a mini-drill is used in a simple pillar stand. Great concentration is required to ensure that the holes are drilled centrally in the pads, so that the holes line up in straight rows and columns. Otherwise, fitting components such as D-connectors and 40-pin ICs will present a major problem.

Larger holes, perhaps 3 mm for mounting bolts or screws, are best not drilled with the mini-drill. More powerful machines, however, tend to pull away at the circuit board material, which gives an unsatisfactory appearance to the holes. It is better to drill first with a 1 mm drill and then use a hand reamer.

High speed steel (HSS) drills can be used for all except fibreglass boards. Hard metal drills are not suitable for fibreglass because the glass embedded in the plastic wears away the cutting edges of the drill so fast that it struggles through the material and does not produce a clean hole. Hard metal drills have a 3 mm or 1/8” thick spindle and a small cutter. These drills, costing upwards of two pounds each, are brittle and break at the slightest provocation. It is an expensive way to make holes.

Drilling under Software Control

The best thing about PCB layout programs is that they automatically generate lists of coordinates of points where holes are to be drilled. If a computer-controlled drilling machine is used, the circuit board simply needs to be fixed to the table and the machine does the rest. The only intervention required is to change drills, which can however be quite tiresome. The program must be stopped, the drilling head must be moved to a specific position, or at least raised away from the board, in order to gain access to the chuck. The height of the drill must be correct, to ensure that the new bit is at the same height as the old one: for this purpose rings can be marked on the drill spindles. Alternatively, drills ready-marked with rings can be bought — but all from the same manufacturer, since different manufacturers put their rings at different heights! And if, after all this effort, the new drill is only used for a few holes, the whole process will have to be repeated just a few seconds later.

Manufacturers soon realised that this was a problem, and developed chucks which can be opened from above using a small handle. This is a great improvement over chucks, such as those frequently found on milling machines and mini-drills, which require a special tool to open them.

Such chucks make a rather expensive addition to a small drilling machine, but for the perfect solution you can expect to pay as much as a thousand pounds: you will need an automatic compressed-air tool changer with tool magazine and a compressor, and everything will work as if by magic. The control program will run a little slower, because the drilling head has to move over to the tool magazine to drop off the old drill and pick up the new one; even here, comfort has its price.

But there is still a big step from the perfect machine to a satisfactorily drilled circuit board. Have you ever considered how registration is preserved through the process of circuit board manufacture? In the PCB layout program the drilling coordinates are known exactly. The positions appear on the film and are then transferred to the circuit board — but where is the reference point?

Getting the film cleanly and accurately aligned with the edges of the circuit board is made impossible by the frequently dirty and roughly-cut base material. The circuit board is generally not rectangular, the material being cut after production using a guillotine.

A computer-controlled drilling machine requires a device that guarantees that the drill lands repeatably at the specified point. This device should be firmly fixed to the machine and in any case designed to make precise alignment easy.

There are many approaches to registration, using try-squares, predrilled holes (a real problem when modifying a circuit), optical registration devices, sticking the film to the circuit board, or ‘intelligent’ motion analysis systems using a camera, TeachIN and coordinate transformations according to reference marks, and many other wonderful methods.

In twenty years of experience developing printed circuit boards in small quantities, the author has found a much simpler solution:

Polar Coordinates!

Here a point on a surface has its position defined not by X- and Y-coordinates as in the Cartesian system, but by a length (distance from a fixed point) and an angle. Polar and Cartesian coordinates can be interconverted without loss of information.

In the PCB layout, draw a circular pad with diameter exactly 3 mm in an unused area; alternatively, use a mounting hole. In the circuit board itself, before exposure, drill a hole with diameter 3.1 mm, in the corresponding place to the pad on the layout. Position the film over the board so that the pad is over the hole. The 0.05 mm wide crack of light around the pad allows exact centring by eye. The centre of the pad is called the reference point. If the unexposed board already has the right dimensions, the film carrying the printed layout can be rotated until its edges are also in alignment with those of the board.

In the layout, as far as possible from the reference point, a second 3 mm diameter pad is drawn. The greatest distance is across the diagonal of the circuit board, but if there is no suitable space, a different place can be chosen. The important thing is that it is far from the reference point. This second point is called the rotation point. Cartesian and polar coordinates are compared in **Figure 1**.

After exposure and etching of the circuit board the reference point will have been etched away, because of the hole previously drilled there. Each other hole to be drilled is at a known distance from the reference hole. This gives us one of the polar coordinates. All we need now is the angle, which is derived from the rotation point. At the rotation point, drill another 3.1 mm diameter hole. This hole must be made extremely accurately, because the position of all the other holes depends on it. It is best to drill first a 1 mm hole and enlarge with a small round file or reamer. The copper of the pad gives a good indication of how well centred the hole is.

All we require now on the table of the computer-controlled drilling machine is a pair of small pegs that fit smoothly and without play in the two holes we have drilled. The peg for the reference point is fixed, while the peg for the rotation point can be slid along a line allowing for various distances between the two points. The coordinates of the fixed peg are known, as is the angle of the sliding rotation-point peg. The drilling data comprise the X- and Y-coordinates of the two points, and so the distance between the two and the angle they make with the Cartesian axes can be calculated using a simple program on the PC. After a translation and a rotation the position of all the other points can be determined, no matter how askew the film was placed on the circuit board during exposure or how the board lies on the CNC table.

This method of registration is suitable for the manufacture of one-off circuit boards, or for a number of different boards. For small-quantity production runs a slightly different procedure can be followed.

Cartesian coordinates: a heresy

Has anyone every seriously wondered why CNC machines always work using the Cartesian coordinate system? Why are X-, Y- and Z-axes always used? Why, when such machines are so difficult to build? The linear guides must be absolutely parallel, because otherwise the carriage will jam. The axes must be at exactly 90 degrees to one another, or else everything goes askew. The table must be absolutely true and the whole machine must be solidly fixed to a base.

These are all disadvantages. But the greatest disadvantage is the linkage between the axes. Consider the X-Y table, the original form of hand-operated milling machine. This has two handwheels, one to move the table in the X-direction, the other to move the table in the Y-direction. There are thus two linear guides, one fixed at an angle of 90 degrees to the other, supporting one another. If the lower mechanism has play in it, this is transferred to the upper one, even if the upper one is absolutely precise. And the lower guide also has to bear the weight of the upper one.

This traditional mechanism stems from a time before computers, when positioning along the axes was controlled manually by a technician using handwheels. Technical drawings are normally marked up with XY coordinates so that the successively required positions can easily be reached by use of the handwheels. In the age of automation the technician is no longer employed and the handwheels are replaced by motors under computer control. But the coordinate system has not changed: in the human imagination everything has a length, a breadth and a height. Curious, when most human actions are polar: ‘take three steps in this direction and then turn right’!

Imagine now how you would drill a circuit board by hand without a pillar drill. With the fingers of one hand you would hold the board steady and with the other hand you would hold the drill. Your drilling arm would be rather higher at the elbow than the other arm, since the mini-drill has a certain height. But you do not move your arms in the X- and Y-directions — no, you turn your drilling arm about the pivot at the elbow and turn and slide the circuit board to suit. You optimise your movements using your visual system — not perfectly, however, as sometimes you might miss a hole hidden by swarf. You do not need a firmly fixed base on which to work; your drilling arm is fixed at the elbow pivot, and what is between this pivot and the circuit board does not matter. Even a small tool between the two makes no difference. Your arms and your sitting position need not be absolutely parallel, or even anywhere near, and there are no 90 degree angles to be seen: two pivots are enough!

The Concept

The system uses two pivots, one for the workpiece (the circuit board) and one for the drilling arm. This allows any desired point on the circuit board to be brought into range on the turntable. This system has the big advantage over a linear construction that only two bearing points are needed whose exact separation is the only quantity that needs to be known. This requires no expensive specialist components: the bearings simply have to remain vertical and free of play. To align an axle to professional standards of accuracy, two so-called taper bearings are used. These can withstand enormous forces and are

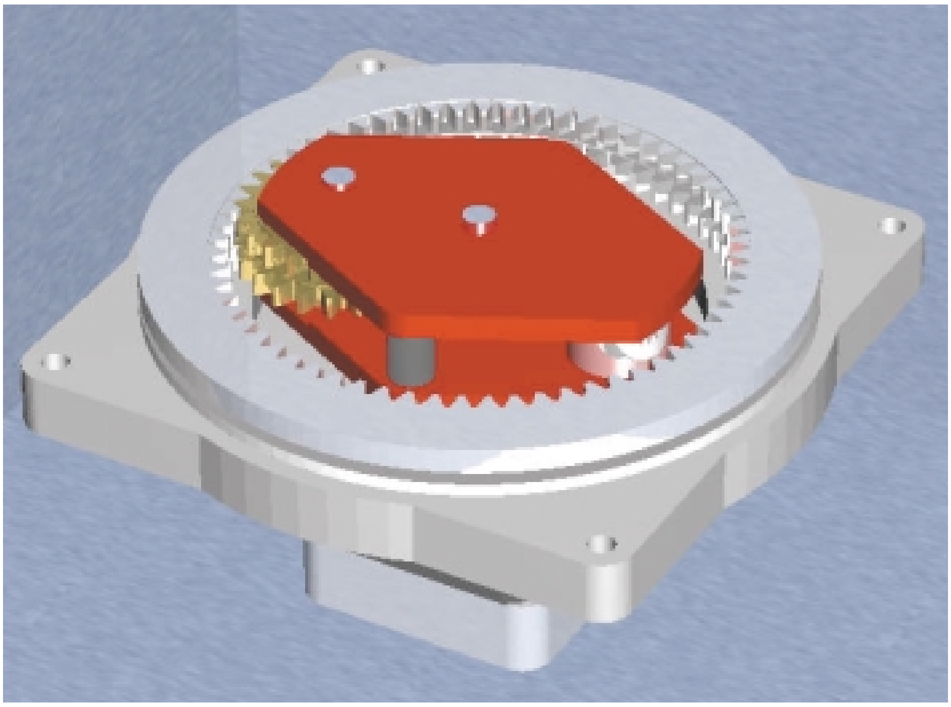


Figure 2. The well-known principle of the ‘axle drive’.

expertly made to remain solid and permanently free of play. This is the main reason why our machine is so economical to build.

A rather significant disadvantage ought to be pointed out. Normally, in conventional linear machines the axles are long threaded rods supported by bearings and turned by a motor. A nut is fixed to the moving part and moves backwards or forwards according to the direction of rotation of the screw. This mechanism naturally gives a high effective gear ratio. Suppose that the screw gives a linear movement of 4 mm per rotation and is driven by a stepper motor with an angular resolution of 200 steps per revolution. The linear motion corresponding to one step is 4/200 = 0.02 mm. That is ideal for this kind of machine. Gearing is therefore completely unnecessary.

Our machine is not driven by a threaded rod, but rather works directly with angular motion. The 240 mm long tool arm has a travel of
240 mm × 2 × 3.14 = 1510 mm
(circumference of circle with arm length as radius)

A stepper motor driving this directly would take 200 steps to make one revolution; the distance corresponding to one step is therefore 1510 mm/200 = 7.55 mm, rather too great for a CNC machine. For a desired resolution of less than 0.04 mm we need to gear the motor down by a factor of at least 7.55 mm/0.04 mm = 190:1. That is not exactly straightforward.

Gearing Mechanism

In the opinion of experts in the field there is no simple gearing mechanism that gives such a high ratio apart from a worm drive. It is mathematically impossible to achieve the ratio with two or three gears without using an unreasonably large number of teeth. At least three stages of gearing down are required with many individual arbors that must each have good quality bearings.

This difficulty, and the unreasonably large amount of play in such a mechanism (and hence problems with repeatability), would cancel out all the advantages of our idea. But you would not have this article in your hands if the problem could not be overcome even despite the firm belief of the experts; and without employing such arcane devices as flexible components fitted between gears with intricately-cut teeth, ball-bearings moving in contorted orbits on even more contorted guides, or belts with different patterns of teeth on either side. If you want to know more, there is plenty to read on this subject on the Internet: you will find ideas there considerably more curious than our drive system. We achieve the desired gearing ratio using four gears. The principle has been known for a very long time, and there is even a VDI (German Association of Engineers) document on the subject (see box); however, practically no-one has yet realised the potential of the idea.

Our drive uses the principle of subtraction. Imagine a travelator such as those found in airports. Our travelator is circular and runs with constant speed. You are running on the travelator in the opposite direction to its rotation; the nearer your speed to that of the travelator, the lower your speed in relation to a fixed point. If you match the speed of the travelator exactly, your net speed is zero: you have a maximum ‘gearing ratio’, the two speeds cancelling one another out.

The situation is similar in our drive system. A spur turns within an annular gear with internal teeth: the actual value of the gearing ratio is not important. Fixed to it is a second spur gear, which turns in a second annular gear. Note that the two spur gears are fixed to the same axle. The two combinations of two gears — i.e., the ratios of spur gear to annual gear — are slightly different from one another, but the more similar they are the greater the overall gearing ratio obtained. Since the second spur gear turns at the rate determined by the first stage rather than that determined by the second, the second annular gear must make a compensating motion. In fact, we subtract the two speeds, just as in the travelator example. This explanation may seem rather complicated, but the illustration in **Figure 2** should make matters clear.

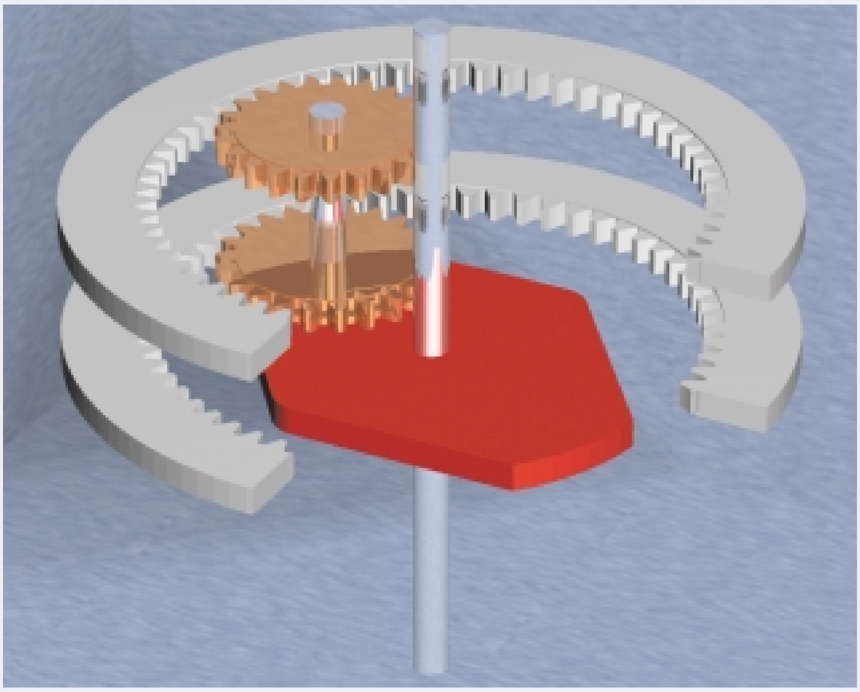
Normally more teeth are required to achieve a higher gearing ratio. Here, however, that is not the case, because it is merely the difference between the two ratios that matters. The author calls this drive system ‘axle drive’ and a patent has been applied for. The chances of success look slim, however, since the design has been known for a long time.

A glance at the internals of the drive system in **Figure 3** immediately suggests many possibilities. For example, there is enough space in the housing to include the parts of a motor; then one would have an incredibly small unit which would be not just a gearbox but a motor with a slowly-turning output shaft with a very high torque. One need only think of the countless applications in vehicles, such as for electric mirror and seat adjusters, or for electric windows, where a flat drive system is required to fit between the door panels. In robots; in component handling systems; this drive is ideal anywhere where a slow but powerful motion is required.

The Drive in Action

Our design not only has constructional advantages over normal linear systems, but also works better in many ways. What happens when the drilling head of a linear machine is at one end of its travel and must move to the opposite end? The head must be moved the entire distance. In our design, things are circular: define 360° as one 'end' of the travel and 0° as the other, and the two are the same! If we are at 360° and need to move to 10°, we do not need to

The drive system: how it works



When we started looking for a simple design for the drive system to give precisely the high gearing ratio required for our machine, we were met by shaking heads: the experts told us that high ratios could only be obtained by using multiple separate stages, each with a small ratio, multiplying together to give the total desired.

Using an ordinary planetary drive a practically-realisable gearing stage with a ratio of up to about 7:1 can be built. This is due to the fact that multiple spur wheels must turn within the annular gear and must clear one another. The diameter of the spur wheels must be less than half that of the annular gear to allow clearance, but must of course not be too small of they will have too few teeth to run smoothly. If higher ratios are required, multiple stages can be cascaded: with two stages we can obtain a ratio of 7x7:1, or 49:1. Add another stage and we have at most 343:1 (although for constructional reasons the maximum obtainable is only just over 300:1). Each stage takes up space, and a three-stage drive with a diameter of 40 mm with attached motor might easily have a total length of over 100 mm.

Further, each stage brings with it play in the mechanism and angular error, which all add together. Economical three-stage planetary drives have a play of about 3 degrees in the output shaft. Although that may sound small, in our design it would translate into an unusable 12 mm of error at the drilling head.

Precision drives in the middle price-bracket — say around one hundred pounds — reduce this play down to just below one degree. For our application, this is unfortunately still completely unusable.

Take a look at the figure and imagine the following sequence of motions.

We will use different tooth counts from those shown in the figure. Suppose the lower annular gear has 45 teeth, its spur wheel has 10 teeth, the upper annular gear has 50 teeth and its spur wheel has 11. These values will provide an excellent example of how the drive works.

The lower annular gear is fixed, and the long main shaft, shown passing through, is free to turn. This is the motor's output shaft, which turns the plate on which the small spur wheels are mounted.

The spur wheel is free to turn on bearings in the plate, and if the main shaft is turned then as the spur wheel runs around the inside of the annular gear, it turns on its own shaft. If the main shaft is turned through 360°, then the spur wheel will have turned on its own axis through 4.5 revolutions since its gearing ratio is 45:10, or 4.5:1. Although the two spur wheels are free to turn relative to the plate they are fixed to one another. So, when the lower spur wheel turns through 4.5 revolutions, the upper one must do so too.

Now if the upper spur wheel, with 11 teeth, turns through 4.5 revolutions, that equates to 49.5 teeth. But the upper spur wheel turning through 49.5 teeth in the upper annular gear causes a problem: the annular gear must make a compensating motion, and the size of this motion must be 50 minus 49.5 teeth, or exactly half a tooth.

Since the upper annular gear, with 50 teeth, moves by a half tooth for each revolution of the main shaft, this means that 100 revolutions of the main shaft are required to obtain an entire revolution. With this combination of 45:10 and 11:50 ratios we have therefore obtained, in one stage, a gearing ratio of 100:1. In our application we use ratios of 60:22 and 23:63, and with no other modifications to the construction obtain a gearing ratio of 231:1. The diagrams show the construction to scale and with exact teeth counts.

In the 19-page VDI (German Association of Engineers) guideline document number 2157 of September 1978, which deals with planetary drives, basic constructional rules and calculation methods are given. The exact arrangement dealt with in that document is called a 'simple planetary drive' which for technical reasons is not suitable for higher gearing ratios.

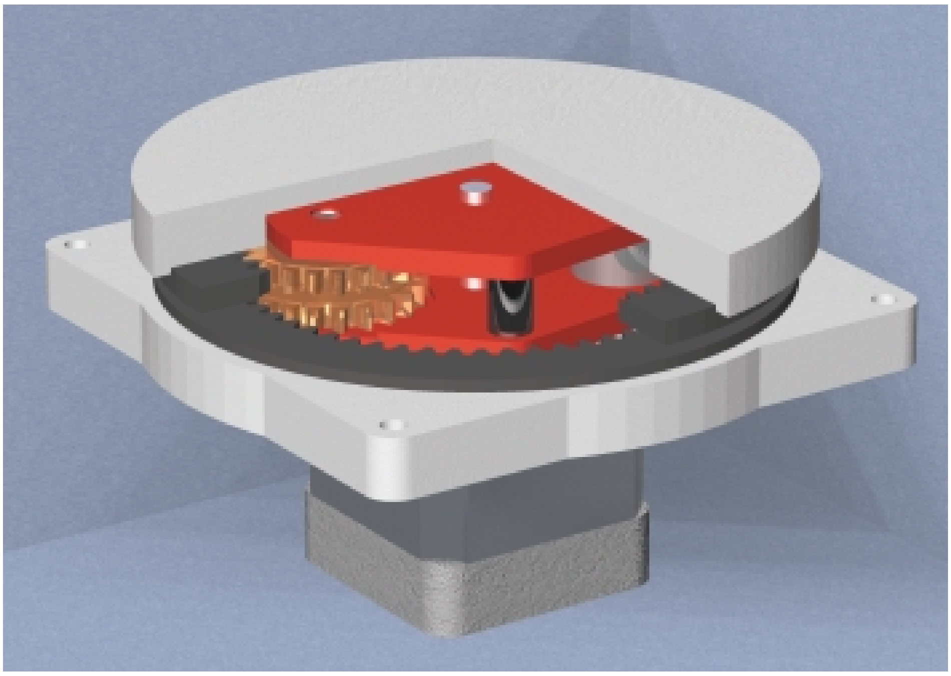


Figure 3. The ‘axle drive’ as used in our machine.

move through 350°, but only through 10°. For this reason our system is four times faster than a conventional CNC machine working over the same area. Also, the mass moved in our system is much less, which allows the machine to work faster still.

The construction of the machine, as can be seen in the photographs (**Figure 4**), aimed a maximum stiffness and minimum weight. The arm and the table can be moved at 70 mm per second, and the stepper motors only require a simple drive circuit because they draw less than 1 A.

Naturally, our machine suffers from the problem of changing the drill bit in the same way as conventional machines as described above. Our design does not provide for automatic tool changing. The construction of a suitable arm would be too complicated and elaborate. But here again we solve this problem in an elegant and surprising way that could not be considered for a conventional machine. There is plenty of space on the drilling table, so we simply add a second arm with a second drilling head. It would of course be possible for the two arms to foul one another, but we can easily arrange things so that one moves out of the way when the other approaches.

The machine is now even faster, since it does not need to wait idly while the tool is changed, does not need to move to a dedicated location to change the tool, and — the clincher — can work with the two arms simultaneously. And further, in order to work with three different drill sizes as is typically required (see above) we can add a third arm. This third arm is, however, not part of the basic design as seen in the cover picture and requires additional parts. The additional bearing housing can be simply screwed to the baseplate. Using three drills simultaneously in a conventional CNC machine would be out of the question.

It is possible to add a fourth arm. Since three drill sizes normally suffice for printed circuit board work, the fourth arm could perhaps carry a different tool such as a router for cutting around the edge of the board. However, the mechanics of the drilling arm are optimised for drilling, and the large sideways forces encountered during milling and routing will adversely affect its accuracy. We will leave the reader in suspense as to whether the author will solve the many problems associated with routing!

(010024-1)

In the next instalment we will be going into more detail concerning the control of the drilling machine from a PC via a microcontroller with an output drive circuit. The command transmission protocol for the microcontroller includes a couple of rather neat ideas!

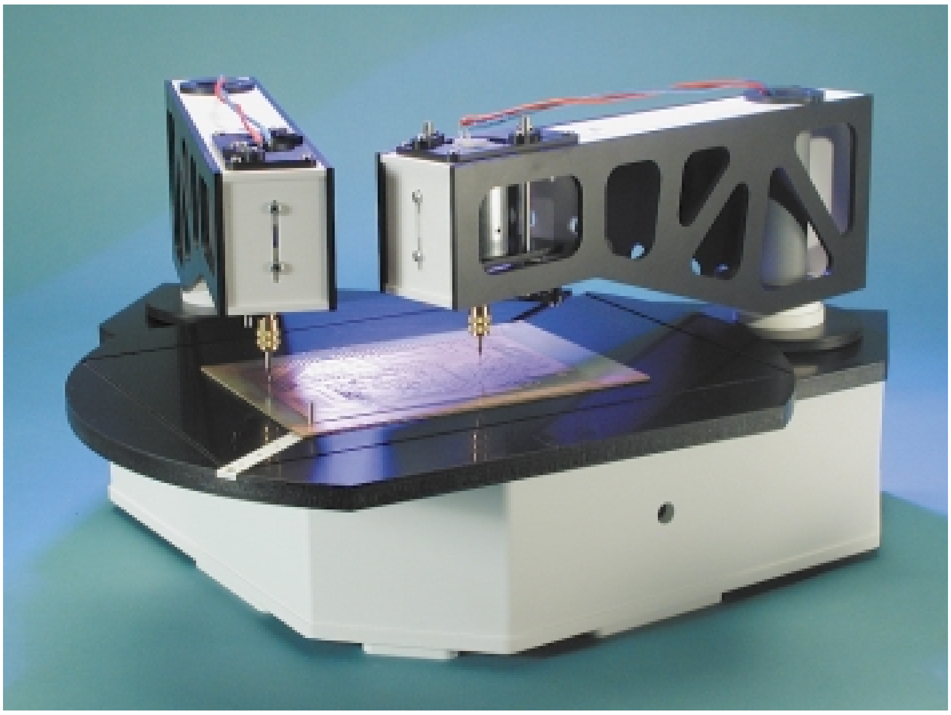


Figure 4. The drilling machine at a glance.

PC Serial Peripheral Design (7)

voltage measurement

By B. Kainka

In previous articles in this series we have described an A/D converter based on a counter. A 'real' A/D converter, however, converts a voltage into a measured value. Usually, this is thought to require an IC; however, as we shall see, a much simpler alternative is available.

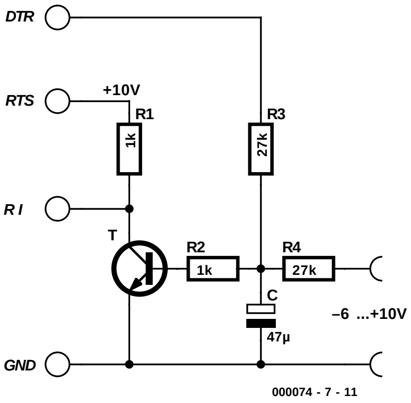


Figure 1. The simple A/D converter

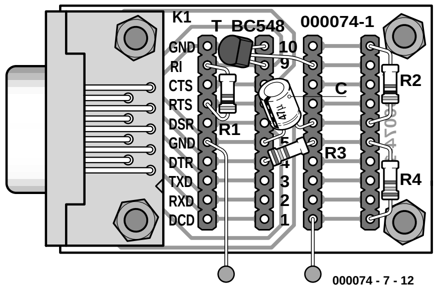


Figure 2. Voltage measurement with the simple A/D converter.

A real A/D converter with a voltage input can be constructed using a single transistor. The transistor in **Figure 1** operates as a comparator, comparing the voltage on the capacitor with a reference value, here 0.7 V. The capacitor is alternately charged and discharged by the DTR output via a resistor, in such a way that the voltage across it is always near to the comparator reference. Depending on the input voltage the output will need to be turned on more or less frequently in order to establish the desired voltage. The count of these events leads to the converted value. **Figure 2** shows how the circuit can be constructed.

A/D converter program

The program in **Listing 1** contains a loop which attempts to keep the voltage across the capacitor as close as possible to the comparator reference voltage. Information on the voltage across the capacitor is available on the RI input: if the voltage lies above the switching threshold of the transistor (about 0.7 V), a collector current flows and the voltage on RI falls. Conversely a lower capacitor voltage turns off the transistor and RI goes high. The measurement loop must drive the voltage on DTR in the opposite direction: when the voltage is too low, DTR must be turned on; when it is too low, DTR must be turned off, setting the output voltage to –10 V. It is important that the switching happens at precisely regular intervals, for example exactly one per millisecond. The number of millisecond periods for which DTR is turned on must be counted. If no input voltage is present, we might expect that the positive and negative states of DTR would be equally common, making the averaged voltage across the capacitor zero. If we look a little more closely, however, we see that we need to consider the threshold voltage of the transistor (about 0.7 V), and so the positive state will be slightly more frequent. With an applied voltage the ratios change. The charging current from the DTR signal must compensate for the charging current from the input.

A negative input voltage leads to more positive DTR states, and con-

Listing 1. Conversion of an 8 bit value

```
Private Sub Form_Load()
 i = OPENCOM(“COM2,1200,N,8,1”)
 If i = 0 Then
    i = OPENCOM(“COM1,1200,N,8,1”)
    Option1.Value = True
 End If
 If i = 0 Then MsgBox (“COM Interface Error”)
 RTS 1
 DTR 1
 Counter1 = 0
 Timer1.Interval = 500
End Sub

Private Sub Form_Unload(Cancel As Integer)
 CLOSECOM
End Sub

Private Sub Option1_Click()
 i = OPENCOM(“COM1,1200,N,8,1”)
 If i = 0 Then MsgBox (“COM1 not available”)
 RTS 1
 DTR 1
End Sub

Private Sub Option2_Click()
 i = OPENCOM(“COM2,1200,N,8,1”)
 If i = 0 Then MsgBox (“COM2 not available”)
 RTS 1
 DTR 1
End Sub

Private Sub Timer1_Timer()
  RTS 1
  DTR 0
  U = 0
  TIMEINIT
  While (RI() = 0) And (TIMEREAD() < 300)
  Wend
  TIMEINIT
  For n = 1 To 255
    If RI() = 1 Then DTR 1 Else DTR 0: U = U + 1
    While TIMEREAD() < n
    Wend
  Next n
DTR 1
  Label1.Caption = Str$(U) + “  “
End Sub
```

versely, a positive voltage leads to more negative DTR states.

The program starts by setting DTR to 1. This is important to ensure that the capacitor is not charged with the wrong polarity. The voltage across the capacitor does not rise above about 1 V, however, because the base-emitter diode in the transistor starts to conduct. In this quiescent state the transistor is fully turned on. In the measurement procedure proper, Timer1.Timer, a new measurement is carried out ever 500 ms. Initially DTR is turned off until RI changes state for the first time. At this point the voltage across the capacitor is equal to the threshold voltage of the transistor. This first loop has a timeout condition to trap errors and runs for at most 300 ms.

The main measurement loop is executed exactly 255 times, during which the number of cases where DTR is low is counted. The measurement always lasts 255 ms and produces 256 different possible results from 0 to 255 (**Figure 3**). This gives the same resolution as an 8-bit A/D converter.

Testing and calibration

Our first test delivered the following results:

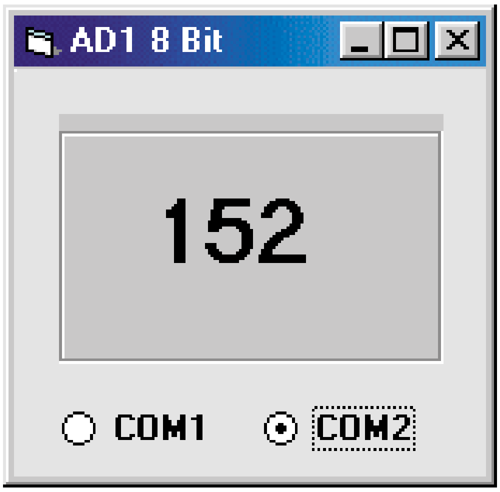


Figure 3. Display of digitised value.

Input	Display
open	118
0 V	110
–3.6 V	68
+3.6 V	152

We can see that an open-circuit input does not display zero, but a rather higher value. Comparing the measurement results for the voltages -3.6 V, 0 V and +3.6 V, we can immediately see that the difference between the negative voltage and zero is 42, as is the difference between the positive voltage and zero. This is encouraging, since it indicates that the converter is linear.

The digital values can now be converted into voltages. The following calculation does the trick:

Voltage = (Display – 110)/11.2

This expression contains the measured zero value and a multiplicative factor giving the number of digital steps per Volt. Both these values will vary from circuit to circuit. It is therefore worthwhile calibrating the circuit. Program AD2.frm (**Listing 2**) provides two slider controls which start off at preset values. The left-hand slider, with a range of 105 to 115 and an initial setting of 110, sets the zero offset. The right-hand slider, with a range of 124 down to 100 and an initial setting of 112, sets the slope. To use the program first short the input leads together and set the zero offset slider appropriately. Then a known voltage must be applied to the input, and the display set to the correct value using the right-hand slider. The A/D converter constructed here uses as a reference the voltage on the DTR output, which is not particularly

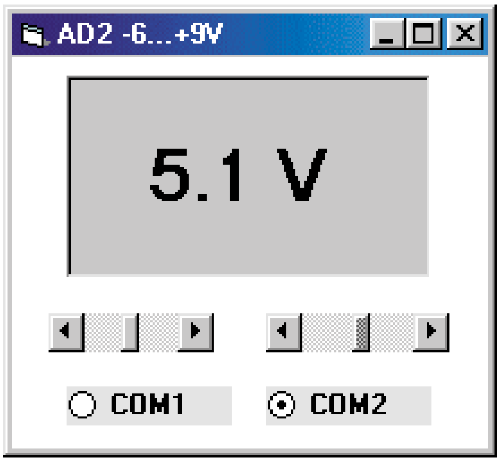


Figure 4. A complete voltmeter.

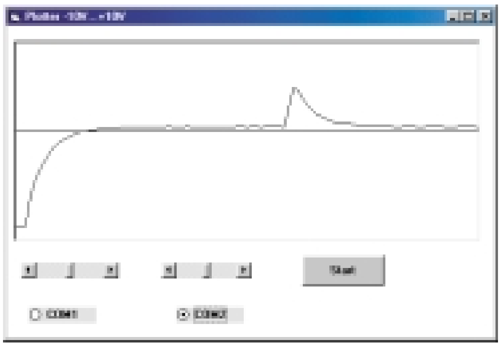


Figure 5. Voltage plotter Plotter2.frm.

reliable. Recalibration is therefore often necessary. **Figure 4** shows the program in action.

This simple A/D converter is reasonably accurate and reliable. The resolution is about 0.1 V, and the measurement range is about –6 V to +9 V. This can already find practical application, for example in testing batteries, and becomes all the more useful when the input voltage can be plotted (**Listing 3**). There are many other applications. **Figure 5** shows the measured voltage across a capacitor being briefly negatively and positively charged. The characteristic exponential discharge curves can be clearly seen.

Hardware improvements

When the possible sources of error in our simple A/D converter are considered, we can see that there are practically no aspects that cannot be improved.

- The basic accuracy depends on the voltage on the serial interface. It would be better to use a proper voltage reference, although that would make the circuit rather more complicated.
- The switching threshold of a comparator constructed from a simple NPN transistor is not 0 V, bit rather about 0.7 V. The exact value depends on the chosen transistor and on temperature. A temperature variation of one degree Celsius changes the threshold by about 2 mV.
- The ratio of the two 27 kΩ resistors affects the measurement; but if 5 % tolerance types are used, the contribution will be dominated by the other sources of error.
- The measurement results do NOT depend on the exact value of the capacitor. This is a particular advantage of this measurement method. Even if a 100 µF capacitor is used in place of the 47 µF capacitor, the results are not affected.

It is relatively straightforward to improve on the design of the comparator (**Figure 6**). Instead of a single transistor, two are used. The pair of NPN transistors forms a differential amplifier, similar to the input stage of an operational amplifier. In this way the DC base-emitter voltage is reduced to a few millivolts. Temperature variation is no longer a problem because the two transistors are affected to the same extent. The total emitter current through the 4.7 kΩ resistor is about 2 mA. When the input voltage is zero the current is

Listing 2. The modified Timer procedure in AD2.frm

```
Private Sub Timer1_Timer()
  RTS 1
  DTR 0
  U = 0
  REALTIME (True)
  TIMEINIT
  While (RI() = 0) And (TIMEREAD() < 300)
  Wend
  TIMEINIT
  For n = 1 To 255
    If RI() = 1 Then DTR 1 Else DTR 0: U = U + 1
    While TIMEREAD() < n
    Wend
  Next n
  REALTIME (False)
  U = (U - HScroll1.Value) / HScroll2.Value * 10
  U = Int(U * 10) / 10
  DTR 1
  Label1.Caption = Str$(U) + “ V”
End Sub
```

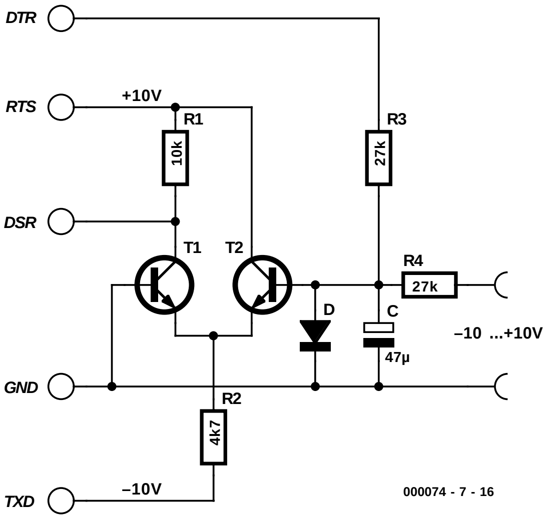


Figure 6. The improved comparator.

Listing 3. Plotter procedures Plotter2.frm

```
Dim y1, y2, x1, x2, n

Private Sub Command1_Click()
 n = 0
End Sub

Private Sub Timer1_Timer()
  RTS 1
  DTR 0
  U = 0
  REALTIME (True)
  TIMEINIT
  While (RI() = 0) And (TIMEREAD() < 300)
  Wend
  TIMEINIT
  For i = 1 To 255
    If RI() = 1 Then DTR 1 Else DTR 0: U = U + 1
    While TIMEREAD() < i
    Wend
  Next i
  REALTIME (False)
  U = (U - HScroll1.Value) / HScroll2.Value * 10
  DTR 1
  y2 = 100 - U * 10
  If n = 0 Then y1 = y2: Picture1.Cls
    x1 = n
  n = n + 5
  x2 = n
  Picture1.Line (x1, y1)-(x2, y2)
  y1 = y2
End Sub
```

divided equally between the two transistors. The voltage drop across the collector resistor of the second transistor is about 10 V, and so the collector voltage remains around zero and close to the switching threshold of the DSR input.

Unlike in the original circuit, the input transistor no longer limits the voltage across the capacitor during intervals between measurements, and so this voltage will rise. For this reason a silicon diode is included to limit the voltage to 0.6 V. The effect of all of these changes is that an open-circuit input gives a reading of zero, and that the measurement range is extended to 10 V with either polarity. **Figure 7** shows the construction of the improved converter.

Software optimisations

Changes are also needed in the software. First it must be taken into account that the DSR input signal is read in the inverted sense: the input is high when the capacitor voltage is above the threshold voltage. The improved characteristics of the measurement circuit also make it worthwhile to increase the accuracy of the measurements. The main loop is now executed 1000 times. **Listing 4** shows the results. With a total input range of 20 V we have a resolution on 0.02 V. As seen in **Figure 8**, a further decimal place can be shown on the display.

(000074-7)

Listing 4. Measurement routine with 1000 quantisation steps

```
Private Sub Timer1_Timer()
  RTS 1
  DTR 0
  U = 0
  REALTIME (True)
  TIMEINIT
  While (DSR() = 1) And (TIMEREAD() < 300)
  Wend
  TIMEINIT
  For n = 1 To 1000
    If DSR() = 0 Then DTR 1 Else DTR 0: U = U + 1
    While TIMEREAD() < n
    Wend
  Next n
  REALTIME (False)
  U = (U - HScroll1.Value) / HScroll2.Value * 10
  U = Int(U * 100) / 100
  DTR 1
  Label1.Caption = Str$(U) + “ V”
End Sub
End Sub
```

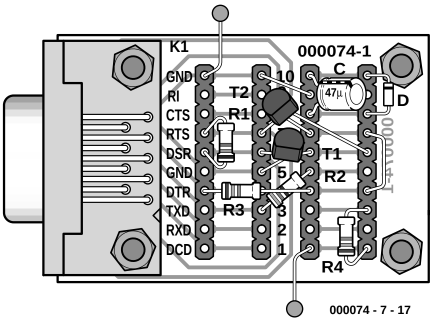
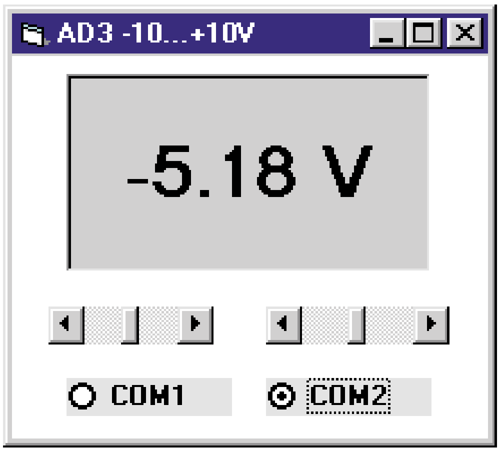


Figure 7. Construction of the improved comparator on the prototyping board.


AD3 -10...+10V

-5.18 V

○ COM1 ● COM2

Figure 8. Result displayed to two decimal places.

MIDI Lights & Slide Control

controlling slide projectors with MIDI commands

Design by A. Mylle

In the January 2001 issue we showed that it was possible to control other things with MIDI besides the usual musical instruments. This time, MIDI signals are used to control light sources. There are four adjustable triac outputs and eight relays to control, for instance, four slide projectors.

Table 1. Types of MIDI-messages.

channel voice messages	channel mode messages	system common messages	system realtime messages	system exclusive messages
note off	all sound off	midi timecode quarter frame	timing clock	system exclusive
note on	reset all controllers	song position pointer	start	sample dump
poly pressure	local control	song select	continue	device inquiry
control change	all notes off	tune request	stop	file dump
program change	omni off	eox (end of exclusive)	active sensing	…
channel pressure	omni on		system reset	midi show control
pitch bend	mono on (poly off)			midi machine control
	poly on (mono off)			…

When controlling lights, it is not easy to maintain synchronisation between light and sound when the control signals reside on different media. There exist a plethora of professional equipment that make synchronisation possible, but usually have a price tag attached that is well beyond the means of the amateur. Cheaper systems may have limited expandability and often cannot be interconnected with systems from a different brand.

Hence the idea to control (slide) projectors and relays via MIDI, where 'light events' can be manipulated with a MIDI/audio-sequencer. Synchronisation is no longer an issue, since everything is now controlled from a PC. By using MIDI as the medium, the application of the MIDI lights is now independent from any hardware or software. If you have at your disposal a MIDI-sequencer with audio capabilities, then it is possible to mix the sampled audio with MIDI. And, if this sequencer possesses more than one separate MIDI-output, then it becomes possible to connect multiple light controllers.

The MIDI solution

This idea is not new. Right from the beginning, with definition V1.0 from the MMA (MIDI Manufacturers Association) were, in the chapter MSC (MIDI Show Control), definitions for protocols related to the control of, for example, lights, stroboscopes, lasers, film projectors, etc. These MSC-messages are 'system-exclusive' commands. These commands have a wide range of applications what immediately means that they are not easily created or edited using a regular sequencer. Besides, an MSC-message for the control of lights is easily 12 bytes long. If these MSC-messages are riding along the same MIDI-interfaces as the MIDI-music then congestion on the MIDI-cable is guaranteed to arise.

More important then the load on the MIDI-cable is the problem that sysex-messages are not normally displayed in graphical form on a sequencer. There will, at best, be a list of hexadecimal numbers in the play list. To edit light events with the music is virtually impossible. There is, however, another possibility.

In **Table 1** is an overview of all the types of MIDI-messages, including a list of the most common.

Normally, commands for light control belong in the MSC-category among the 'system exclusive messages'. However, if we look at the list 'channel voice messages' to 'control change', then these messages consist of only 3 bytes: one status byte and 2 data bytes. Such a control change usually serves the purpose of altering the parameters of the sound source, such as, for example, volume, reverberation, panning, etc.

Most of the possible 128 control changes (refer to sidebar 'MIDI-bytes') are already allocated fixed functions. But, there are also a few that are free to be used for equipment

Table 2. Utilised controllers.

General purpose continuous controller #1	ctrl 16	brightness projector #1
General purpose continuous controller #2	ctrl 17	brightness projector #2
General purpose continuous controller #3	ctrl 18	brightness projector #3
General purpose continuous controller #4	ctrl 19	brightness projector #4
General purpose switch controller #5	ctrl 80	relay projector #1
General purpose switch controller #6	ctrl 81	relay projector #2
General purpose switch controller #7	ctrl 82	relay projector #3
General purpose switch controller #8	ctrl 83	relay projector #4
undefined controller	ctrl 84	extra relay #1
undefined controller	ctrl 85	extra relay #2
undefined controller	ctrl 86	extra relay #3
undefined controller	ctrl 87	extra relay #4
reset all controllers	ctrl 121	all lamps and relays off
'running status' is recognised (see text)		

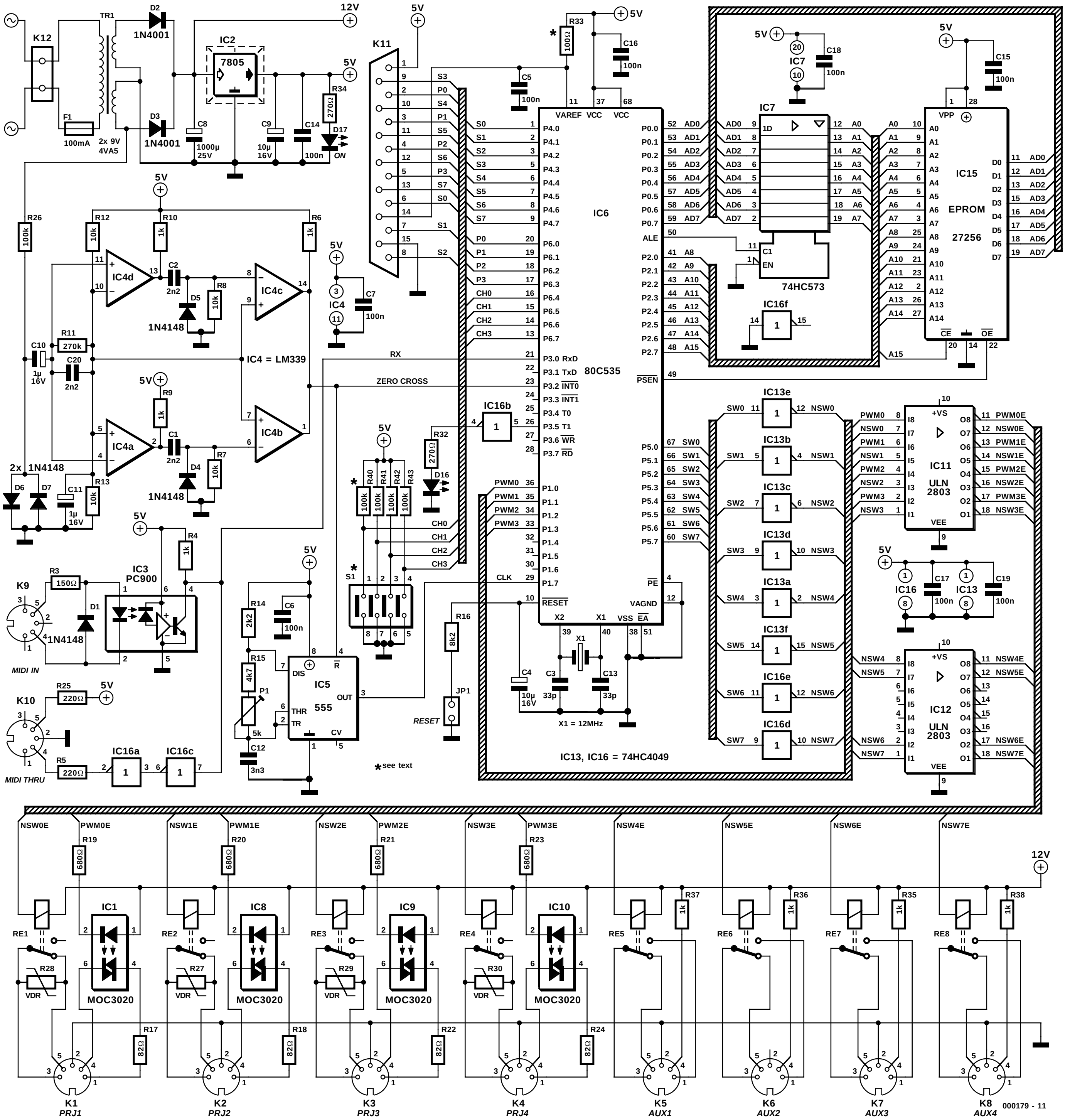


Figure 1. Schematic of the MIDI-light-control.

specific functions, for instance. If we select our light control commands from these available control changes then, suddenly, everything becomes much simpler. Lights may now be manipulated in the same way as the modulation of volume. A nice feature is that it becomes possible to program the sequencer to operate a light with the modulation wheel and switch a relay with the sustain pedal. Providing the light module with a MIDI-through-connection means that it can simply be connected in series with the sound module using a standard MIDI cable. Of course, there is the prerequisite that the sound module does not respond to the controller commands used by the light module. These controller commands have been chosen such that the risk of interference is virtually eliminated. The MIDI-light-module uses only a couple of controllers from channel 16 and since MIDI files seldom contain these specific controllers there is unlikely to be a problem. The MIDI-light-controller can drive 4 projectors and 8 relays when

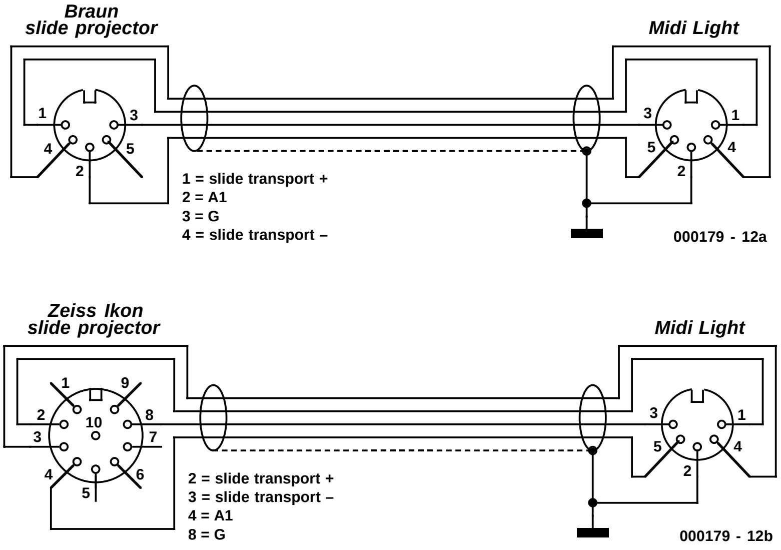


Figure 2. Connector details of the control terminals of slide projectors from Braun and Zeiss Ikon.

used with the MIDI-controllers summarised in **Table 2**.

Possibilities

Control of 4 projectors

The light intensity is controlled by the value of the controller. Every change in intensity requires an event. Value ‘0’ means the light is off and value ‘127’ means light is on maximum. The pre-glow is adjustable and is identical for all four channels. The lamp characteristics are linearised.

Control of 4 normally open contacts

(To advance slides, for example)
The relay is controlled in real time by the value of the controller. Every change of state requires an event. A value of ‘0’ means the contacts are open, a value of ‘127’ means contacts closed. This means that the user defines the contact closure time.

Driving 4 relays

(Make and break contacts can be utilised for a variety of purposes)
The relay is controlled in real time by the value of the controller. Every state change requires an event. When the value is ‘0’ the relay is not energised, when the value is ‘127’ the relay is energised. The DIN connector makes both the make- and break contact available together with a 12 V signal via a series resistor. This approach makes the interface more versatile. Either use the contact to switch low voltage directly or use an opto-triac to switch 230 V. This last suggestion can be implemented by combining an opto-triac with a power triac and fitting this in an enclosure with built-in plug and socket. This way everything is safely separated from the 230 V mains. Never use the relay contacts to switch 230 V directly!

The hardware

The light module (**Figure 1**) is built around an 80C535-12 from Infineon (Siemens). It is a member of the 8051-family with, among other things, four pulse width modulators built around timer 2, 7 I/O ports and an 8 channel A/D-converter. The PWM section performs the phase control of the mains voltage for the 4 projector lamps.

The PWM outputs each trigger an opto-triac type MOC3020 (IC1, IC8, IC9 and IC10), which in turn fires the power triac internal to the projector. If a modification to the projector is necessary, it is limited to the fitting of a triac. If the projector has a built-in brightness control it will be sufficient to make the anode- and gate connections available on the outside. A number of common projector connections are shown in **Figure 2**.

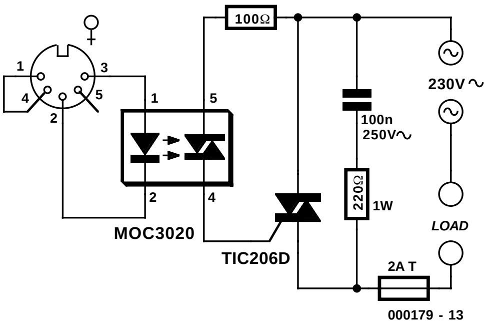
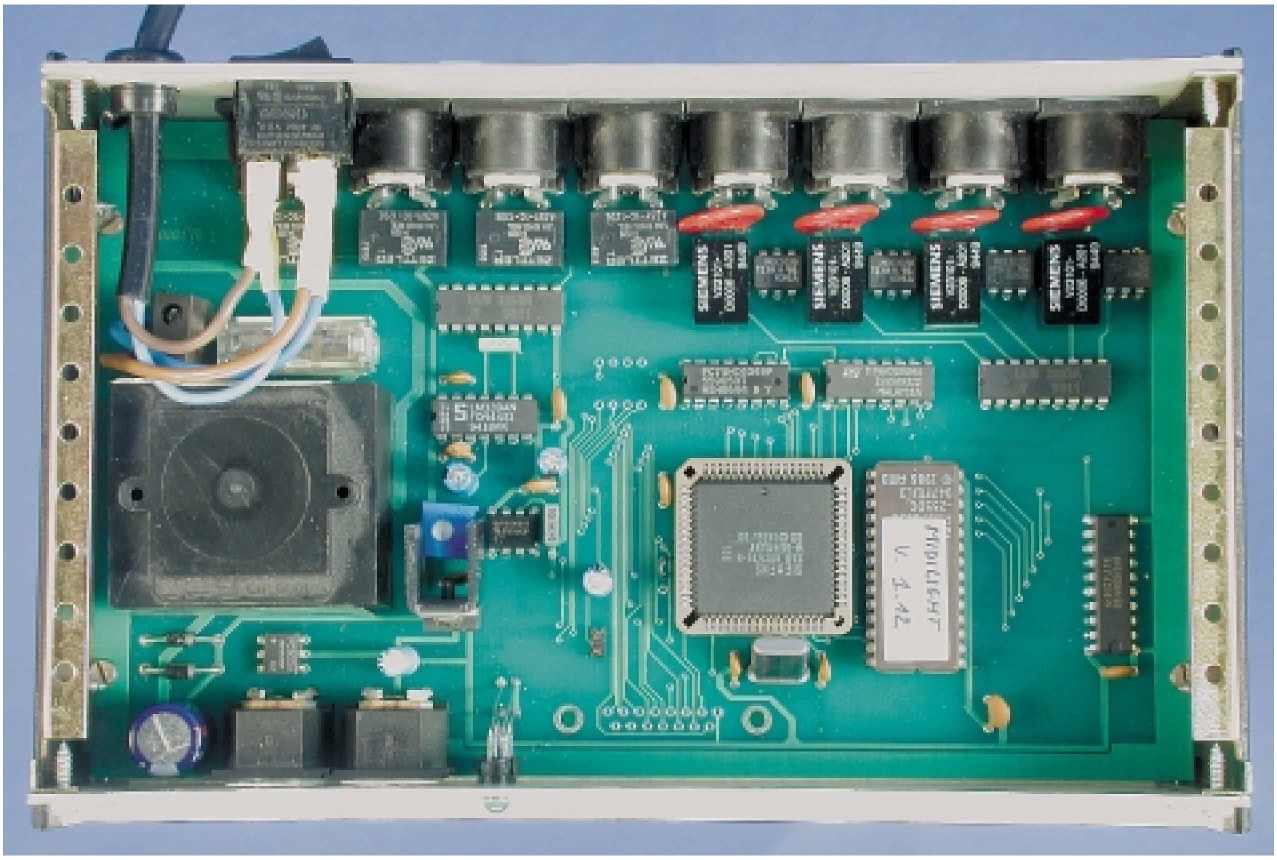


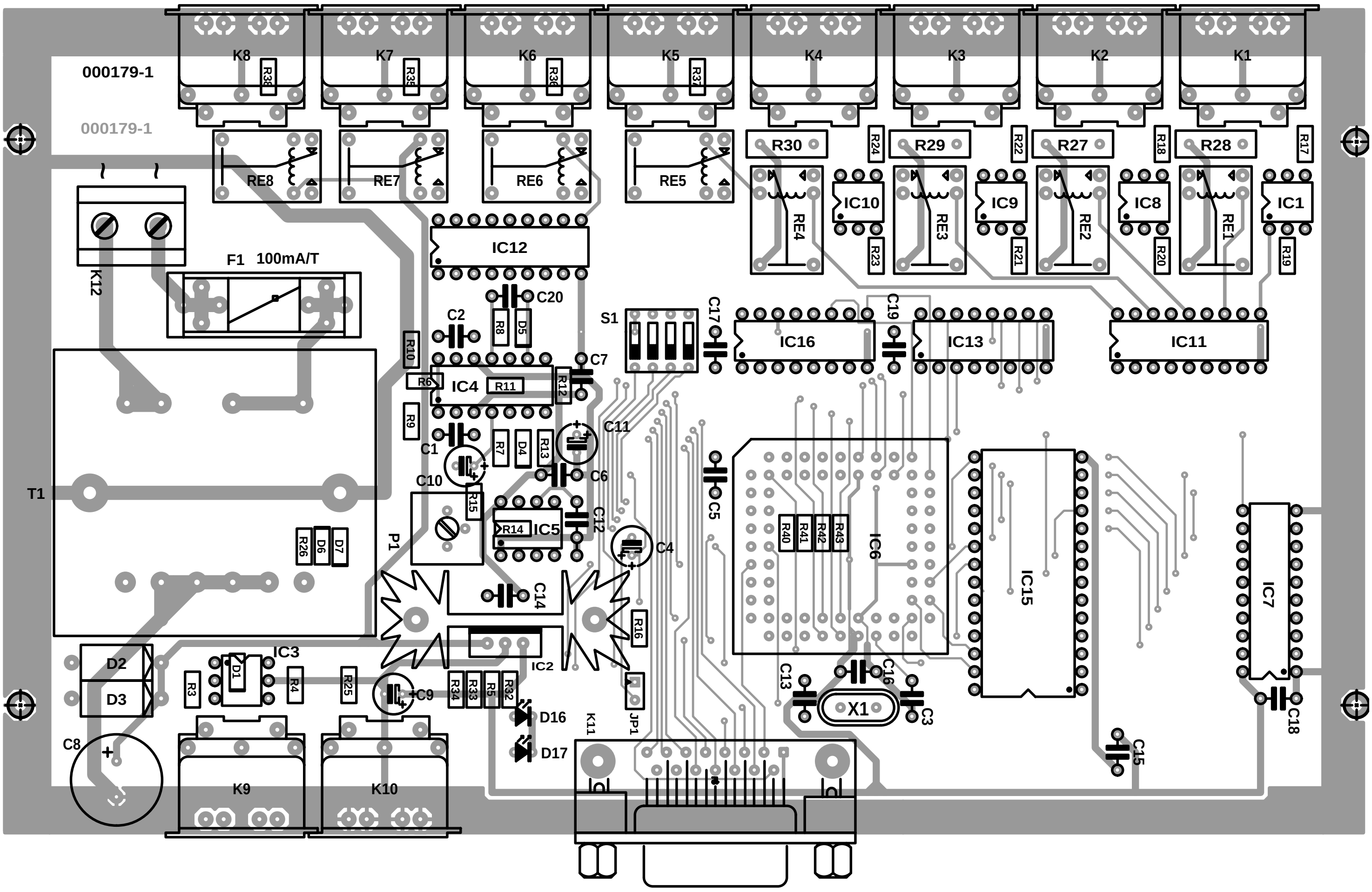
Figure 3. Schematic of a circuit to control 230 V load (max. 2 A).

To obtain good mains synchronisation of the pulse width modulators, a zero-crossing-detector is built around an LM339 quad-comparator (IC4) which, every 10 ms, activates interrupt 0 on pin 32. Careful attention was paid to the design of this detector to avoid interference to the light control from the 1350 Hz time code signal that rides on top of the 50 Hz mains voltage. The AC signal is tapped from the (highest available) voltage on the secondary of the mains transformer. On the outputs of the first two comparators, alternating between them, appears a falling

Running status

MIDI-commands (such as ‘control change’) consist always of a single status byte followed by one or more data bytes. The status byte defines the type of command that is to follow, while the data bytes contain the actual information. In the case of the control change, for example, the status byte is followed by 2 more data bytes. The first contains the controller number and the second contains the controller value. When the next MIDI-message is intended for the same MIDI-channel, it is not necessary to repeat the status byte. This status byte would have been identical to the previous one. Such successive MIDI-commands of the same type and for the same channel require only a single status byte, sent at the beginning of the set of messages. A status byte is only transmitted if it differs from the previous one. This entire mechanism is called ‘running status’. Some older MIDI equipment is unable to handle the running status. The better sequencers permit this mode to be switched off.




000179-1
000179-1
K8
K7
K6
K5
K4
K3
K2
K1
R38
R35
R36
R37
RE8
RE7
RE6
RE5
R30
R24
R29
R22
R27
R18
R28
R17
RE4
IC10
RE3
IC9
RE2
IC8
RE1
IC1
R23
R21
R20
R19
K12
F1
100mA/T
IC12
C20
C2
R8
D5
R10
S1
C17
IC16
C19
IC13
IC11
C7
R6
IC4
R11
R12
R9
C11
C1
R7
D4
R13
C6
C10
R15
C5
IC5
R14
C12
P1
T1
R26
D6
D7
R40
R41
R42
R43
IC6
IC15
IC7
C4
C14
R16
IC2
D2
D3
IC3
D1
R3
R4
R25
C9
R34
R33
R5
R32
D16
D17
K11
JP1
C13
C16
X1
C3
C18
C15
C8
K9
K10

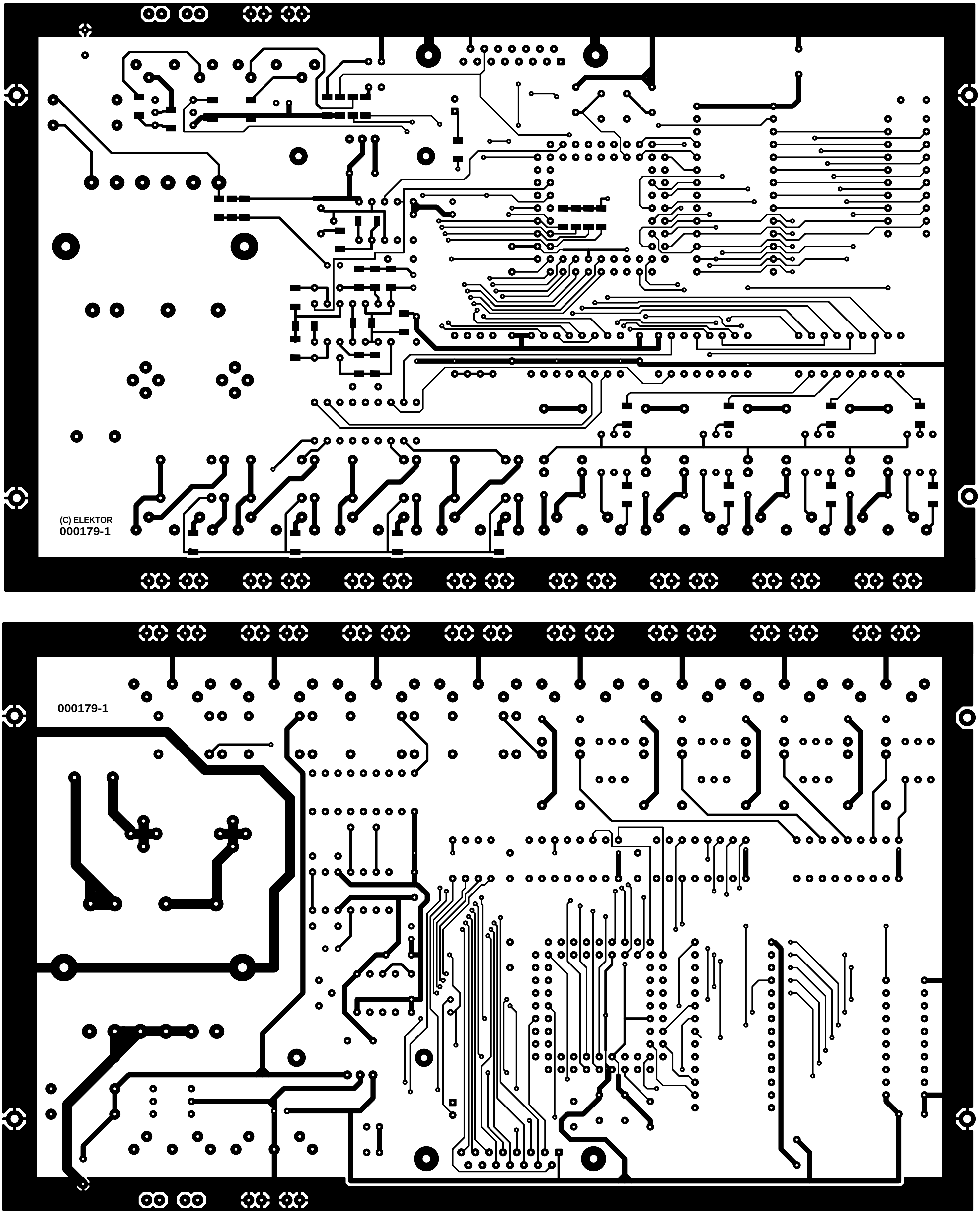


Figure 4. The PCB is double sided. It also contains the mains transformer and all connectors.

edge at the instance of each zero-crossing. Each falling edge is converted to a short pulse of a few microseconds duration using an RC-network and a second pair of comparators. This pulse is sufficiently long to be detected by the interrupt routine. As a consequence, this routine is executed every 10 ms.

Because the zero-crossing is detected at the secondary winding of the transformer in the module itself and not the projector, the actual zero-crossing of the lamp may not coincide exactly with the pulse from the detector. You will not notice this in practice, provided both the module and the projector are powered from the same phase.

The PWM-clock of about 28 kHz is generated by IC5, a 555. This signal is reset at every zero-crossing. This is necessary to ensure that every half mains cycle is started with a whole pulse. If this were not done, there would be small fluctuations in the brightness of the lamp because the PWM-clock is not an integer multiple of the mains frequency.

The MIDI-IN-interface is, as prescribed in the MIDI standard, not much more than an opto-coupler PC900 (IC3) connected to the serial port of the microcontroller. The MIDI-input is also immediately looped to the MIDI-output, which enables the light module to be connected in series with existing equipment. LED D16 lights up briefly when the module has received signals that control the projectors or relays.

Relays Re1-Re8 are driven from port 5 via buffers IC13 and IC16 and relay drivers IC11 and IC12.

The contacts of relays Re1-Re4 are suppressed with VDRs (R27-R30).

Latch IC7 filters, with the aid of the ALE signal, the least significant byte of the address bus from the multiplexed address/data bus for use by the EEPROM.

The software

This comprises 3 parts:

The zero-crossing detector triggers the INT0 routine, which performs the phase control by reading the CC-registers.

The serial input places all received MIDI-bytes in a buffer, 128 bytes long.

The main program reads the bytes from the buffer and processes the relevant commands. It detects the running status, drives the relays and refreshes the CC-registers.

Phase control is obtained as follows: The pulse from the zero-crossing detector triggers IRQ0 every 10 ms – one half mains cycle – which sets timer 2 to a value of 65535 (FFFFh). The first PWM clock pulse causes the timer to overflow which then counts up from zero. As a consequence of this overflow, all PWM outputs are set to '0'. All this occurs immediately after the zero-crossing. The counter continues to count at the rate of the PWM clock, which generates 256 pulses in 10 ms (see

MIDI-bytes

With MIDI, the MSB defines if it is a status byte (MSB = '1') or a data byte (MSB = '0'). This leaves 7 bit values for the data bytes. With control changes, the first byte contains the number of the controller and the second contains the actual controller value. If a higher resolution is required, such as for 'pitch bend', then an extra byte is sent for the controller value, or as with some controllers, a separate MSB- and LSB-controller is used. There are a maximum of 128 controllers that may be operated. Fortunately, the existence of MIDI-standards such as GM, GS and XG ensures that, among other things, a certain controller number on different pieces of equipment has the same function. In this way MIDI-files sound almost identical on sound sources from different manufacturers.

COMPONENTS LIST

Resistors:

(All resistors SMD case # 1206)
R1,R2,R31 = not fitted
R3 = 150Ω
R4,R6,R9,R10,R35-R38 = 1kΩ
R5,R25 = 220Ω
R7,R8,R12,R13 = 10kΩ
R11 = 270kΩ
R14 = 2kΩ2
R15 = 4kΩ7
R16 = 8kΩ2
R17,R18,R22,R24 = 82Ω
R19-R21,R23 = 680Ω
R26,R40...R43* = 100kΩ
R27-R30 = VDR 48-96 V, e.g., Siemens S10K50 or S07K50
R32,R34 = 270 Ω
R33 = 100Ω*
P1 = 5kΩ preset, 'ordinary' vertical model or Bourns type 3386

Capacitors:

C1,C2,C20 = 2nF2
C3,C13 = 33pF
C4,C9 = 10μF 16V radial
C5,C6,C7,C14-C19 = 100nF, lead pitch 5mm
C8 = 1000μF 25V radial
C10,C11 = 1μF 16V radial
C12 = 3nF3
C14 = 100pF

Semiconductors:

D1,D4-D7 = LL4148 (SMD)
D2,D3 = 1N4001
D16 = LED, green
D17 = LED, yellow
IC1,IC8,IC9,IC10 = MOC3020 (QT, formerly Motorola: www.qtopto.com)
IC2 = 7805
IC3 = PC900 (Sharp)
IC4 = LM339
IC5 = 555
IC6 = SAB80C535-N
IC7 = 74HC573
IC11,IC12 = ULN2803
IC13,IC16 = 74HC4049
IC15 = 27C256, programmed, order code **000179-21**

Miscellaneous:

K1-K10 = 5-way DIN socket, PCB mount, pjns arranged at 180°
K11 = 15-way sub-D socket (female), PCB mount, angled pins*
K12 = 2-way PCB terminal block, lead pitch 7.5 mm
X1 = 12 MHz quartz crystal
F1 = fuse, 100 mAT (time lag), with PCB mount holder
S1 = 4-way DIP switch
T1 = mains transformer, sec. 2 x 9 V/4.5VA (e.g., Monacor/Monarch VTR4209)
JP1 = 2-way pinheader
68-pin PLCC socket
Heatsink for IC2, e.g., SK104-50 (Fischer, Dau Components)
Re1-Re8 = 12-V relay (e.g., Zettler AZ5Y-1C-12DE or Siemens V23101-D6-A201
Enclosure type LC650 (Eurodis, Bolton)
PCB, order code **000179-1** (see Readers Services page and EE website)
Disk, contains sourcecode and binary file, order code **000179-11** (see Readers Services page and EE website)
* optional, for extensions

Generating Controller Events using Cubasis AV

It is relatively easy, using the program Cubasis from Steinberg to generate controller commands. Start by setting the time base. If you start with an existing MIDI-file than this is already fixed, of course. But if you use only audio, then choose this yourself. The time axis is divided in bars. The number of bars that are played per second are: Tempo x measure / 60.
Example: 120 bpm (beats per minute) and 2/4 equals one bar per second. Pay attention, the first bar is '1'. At bar number 10, 9 seconds have elapsed.
It is easiest to create 'parts' for each series of controllers. For example, make parts for long fade in, short fade in, fade out, change, etc. Every part has a name. Each part contains the controller events.

A controller event is created as follows:

Define the length of a part and set markers at the desired beginning and end.

Select the active track on the left and set this to MIDI-channel 16 of the MIDI-out interface.

Double click on the empty space between the markers at the height of the track. There appears an empty part.

Select this part.

Click 'Edit – List' and in the pull down list 'ins. control change'.

On the right of the screen, select the pencil using the right mouse button and mark a first event at the beginning of the part.

Change the list 'val. 1' to the desired controller number. This is '15' for the first channel. In the status field appears the name of the controller 'General Purpose 1'. The brightness 'val. 2' is from 0 (off) to 127 (max).

Click 'Edit – Edit'. At bottom left there is now controller number 15. Don't click it, because the pull down menu does not contain all 127 controllers. For other controllers it only works when starting from a selected controller in 'List Edit'.

In the bottom part of the screen, click either the cross or the pencil wilt the right mouse button. With the alt key held down, draw new events by dragging the starting value of the end value. The resolution can be changed using the 'snap'-parameter.

Parts for advancing slides consist of first sending a value of 127 followed by 0. The time between these events can be defined and on some projectors this is used to go forwards or backwards.

Once you have a 'library' of a number of basic parts, most of the hard work has been done, you can now simply copy and paste.

below). Every PWM-output has a 'capture&compare'-register. The register for each channel contains a value that corresponds to the desired phase angle. When the counter value of timer 2 is equal to the value in the CC-register, the corresponding PWM-output becomes '1' and triggers the triac. By varying the value in the CC registers, the phase angle of each channel can be controlled from lamp off (CC equals FFFFh) to lamp fully on (CC equals zero).

The CCs are 16-bit registers while the MIDI-controls are only 128 values (7 bits). We vary only the LSB (MSB=FFh) such that the values 0-127 from the MIDI-controller are translated to 255…0 using a table. Making the table slightly non-linear allows the lamp characteristic to be linearised. When adjusting the brightness of the lamp from zero to maximum it appears that all the steps are equally spaced across the whole range.

Setting the PWM-clock to slightly more than 256 pulses, for example 280 pulses per 10 ms (=28 kHz), causes the counter to trip a little before the next zero crossing, when set to minimum brightness. As a consequence, the lamp is not completely off at the minimum setting. This pre-glow ensures that the lamp reacts faster when fading in. Because this adjustment is part of the PWM-clock, it is the same for all lamps.

Except for reloading timer 2 during the interrupt routine for the phase control, the involvement of the microcontroller is limited to refreshing the CC-registers. It is imperative that this happens at the zero crossing. If this is not the case then the lamp may flicker because the new value may be smaller than the current counter value.

Incoming MIDI-bytes are put into a 128 byte long buffer by the serial interrupt routine. The main program processes the bytes in the buffer.

Construction and Adjustment

A double-sided PCB was designed for this circuit. It includes the mains transformer and all connectors (**Figure 3**).

When purchasing the PLCC68 socket for the microcontroller, check that it suits the PCB since multiple pin configurations exist. The construction should otherwise be straightforward. It is best if you mount the components in the following order: passive components, the diodes, the IC sockets, the relay and finally the connectors. Mount the ICs only after you have checked the power supply voltages. With the exception of IC1 and IC3, it is possible to fit the other ICs without a socket. The PCB layout allows for several types of mains transformer. Make sure that the primary side of the transformer is on the same side as the fuse.

Adjust P1 to the centre position and connect the projectors to K2-K5. In the event of a three-phase 230 V mains, ensure that all the projectors are connected to the same phase. As a check, LED D16 should light briefly after switching the power supply on (this also happens after a manual reset, using jumper J1). Turn P1 until the lamps just show a dim red glow.

This concludes the adjustments. You can now send MIDI signals to control the module. Of course, the intricate details of how that is done depend on the sequencer you use. The LED will light briefly every time a control message has been received.

(000179-1)

A few useful MIDI-links

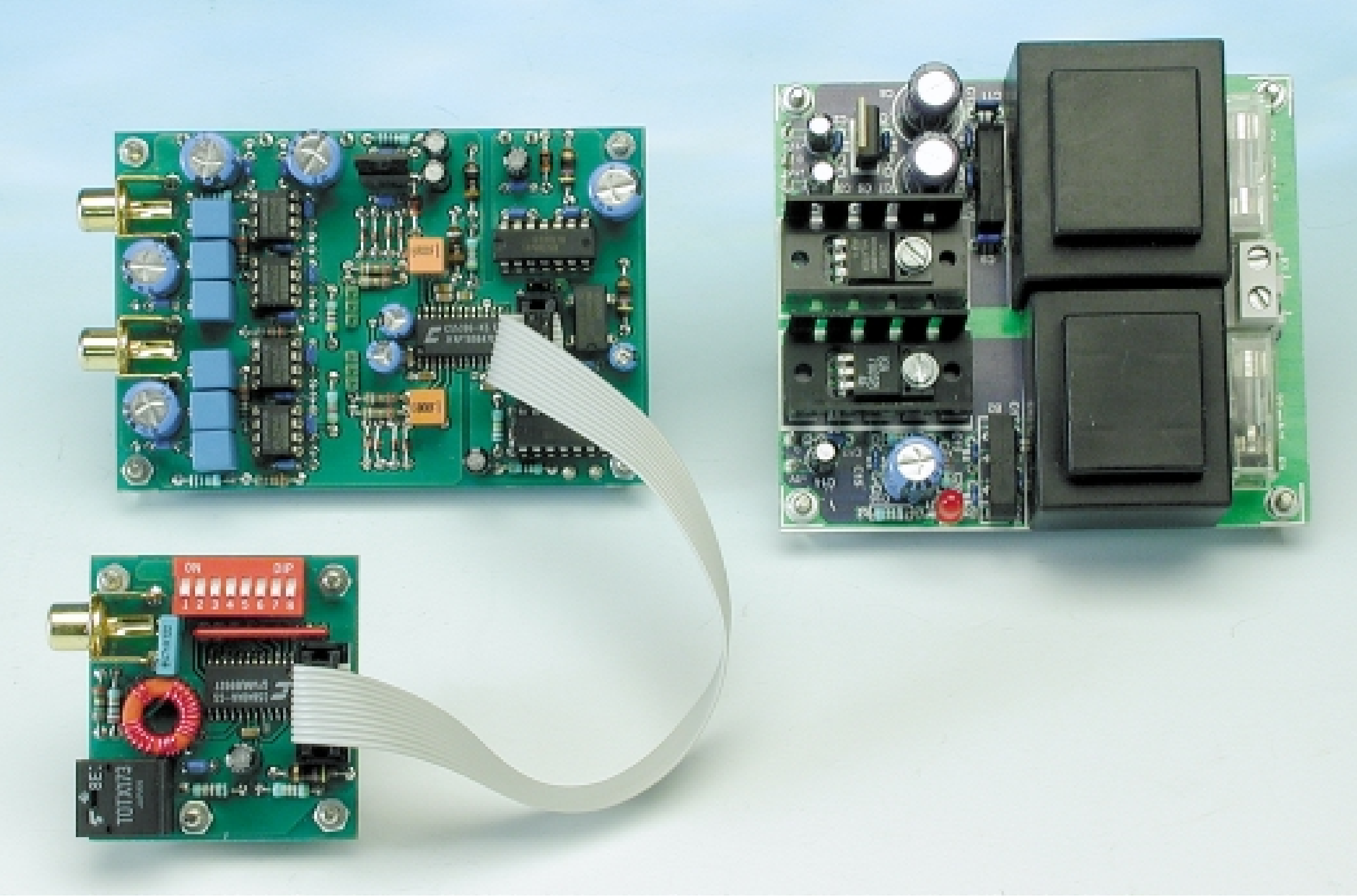
The MIDI Farm Internet:
http://www.midifarm.com/info/
MMA home page at www.midi.org:
http://www.midi.org/
Harmony Central: MIDI Tools and Resources:
http://www.harmony-central.com/MIDI/

Audio ADC 2001

24-bit, 96 kHz sampling

Design by T. Giesberts

It is barely four years since we presented our 20-bit A/D Converter as the newest of the new. However, development continues at a rapid pace. At the present moment, 24-bit, 96-kHz sampling is the standard. It’s thus high time for a successor that is completely up to date!



Anyone who is a bit familiar with the subject will immediately see that the A/D converter described here is almost the twin of the 20-bit design of four years ago. The only difference is that the resolution and speed have been increased by the use of more recent components, so that the present design is fully state of the art. This circuit can thus be regarded as a sort of ‘Mark II’ version of the previous circuit.

Once again, an integrated A/D converter made by Crystal, the CS5396, occupies the central position in the design. This successor to the ‘old’ CS5390 converter, which was used in the previous design, has been specially developed for 24-bit A/D conversion of stereo audio signals with sampling rates up to 100 kHz. The CS5396 employs a seventh-order, tri-level delta–sigma modulator and a digital filter, which makes an external anti-aliasing filter unnecessary. The differential architecture produces exceptionally good noise rejection: the maximum signal/noise ratio of the IC is 105 dB and the dynamic range is no less than 120 dB! We have intentionally chosen the CS5396 for this design, instead of the closely related CS5397, since the former type has a linear-phase filter that is optimised for audio applications. It has a pass-band ripple of only ±0.005 dB and a stopband rejection of >117 dB.

On the output side, we also find a pin-compatible successor to the previously used IC. Here the CS8402 has given way to the brand new CS8404A, which Crystal designates as a '96 kHz digital audio transmitter'. This IC is especially intended to be used for coding and transferring audio data according to one of the well known interface standards (AES/EBU, IEC958, S/PDIF or EIAJ CP-340). Inside the CS8404A, the digital audio data are multiplexed and encoded before being sent to the output. In addition to the S/PDIF output, which is electrically isolated by a transformer, the circuit also has an optical output.

Input section

The input stage of the converter has been intentionally simplified in comparison to the 'old' 20-bit design, with the result that it can process only asymmetric signals. This means that it is now possible to place Cinch sockets on the printed circuit board, which reduces the likelihood of interference problems.

Now let's look at **Figure 1**, which shows the complete schematic diagram of the Audio ADC 2000. For each channel, the input signal is first DC decoupled, since the converter has an asymmetric power supply and each input stage thus has an input offset. VCOM is used to set the input stage to the proper voltage. For this purpose, VCOM is supplied to the opamps by separate decoupling networks (R4/C4/C5 and R10/C10/C11).

The CS5396 has one symmetric input

Measured results

A few measurements have been made in the digital domain using a special measurement system from Crystal ('CDBCapture+ board'). In such cases, the signal from the optical output has been sampled and further processed.

The first figure, **Figure A**, shows a FFT analysis of 16,384 samples taken from the left channel with the converter driven to just about full scale. The THD+noise is here more than 104 dB down. Measurements on the right channel show similar results, with the THD+noise lying below –105 dB.

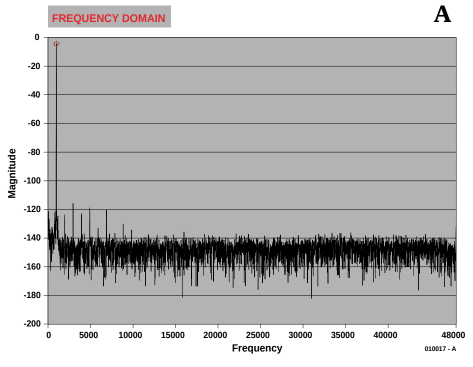


Figure B shows a FFT of a 20-kHz signal at full drive. The only harmonic that can be seen is a second harmonic at –114 dB, and there are also a few interference products (probably crosstalk from the digital part of the converter), but these lie below –120 dB! The right channel was clean, with only a second harmonic at –123 dB.

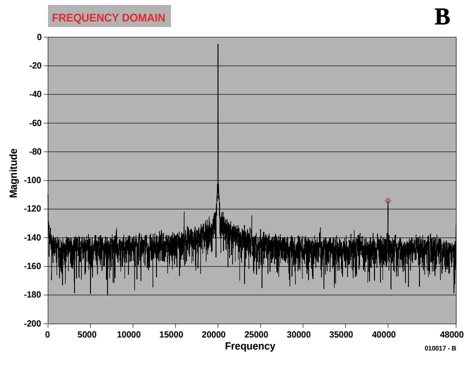
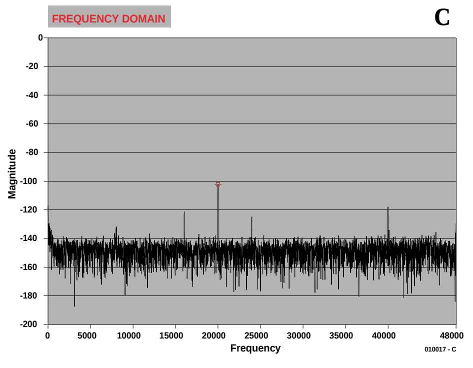
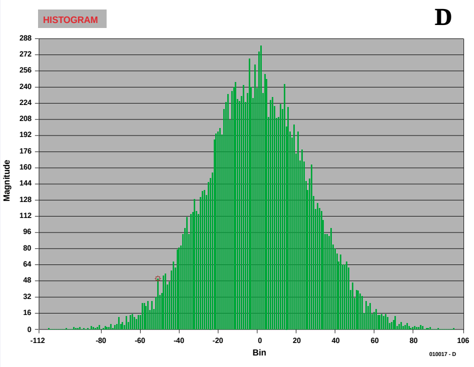


Figure C shows an FFT of the crosstalk from the right channel to the left channel for a 20-kHz signal with full drive. The channel separation here is greater than 102 dB. In the other direction (from left to right), this is 96 dB, but in the latter case only the 20-kHz component is visible.



Finally, **Figure D** presents a simple statistical measurement in the form of a histogram showing the distribution of the output values with the input shorted. This is a 'normal' distribution that shows a regular Gaussian shape.



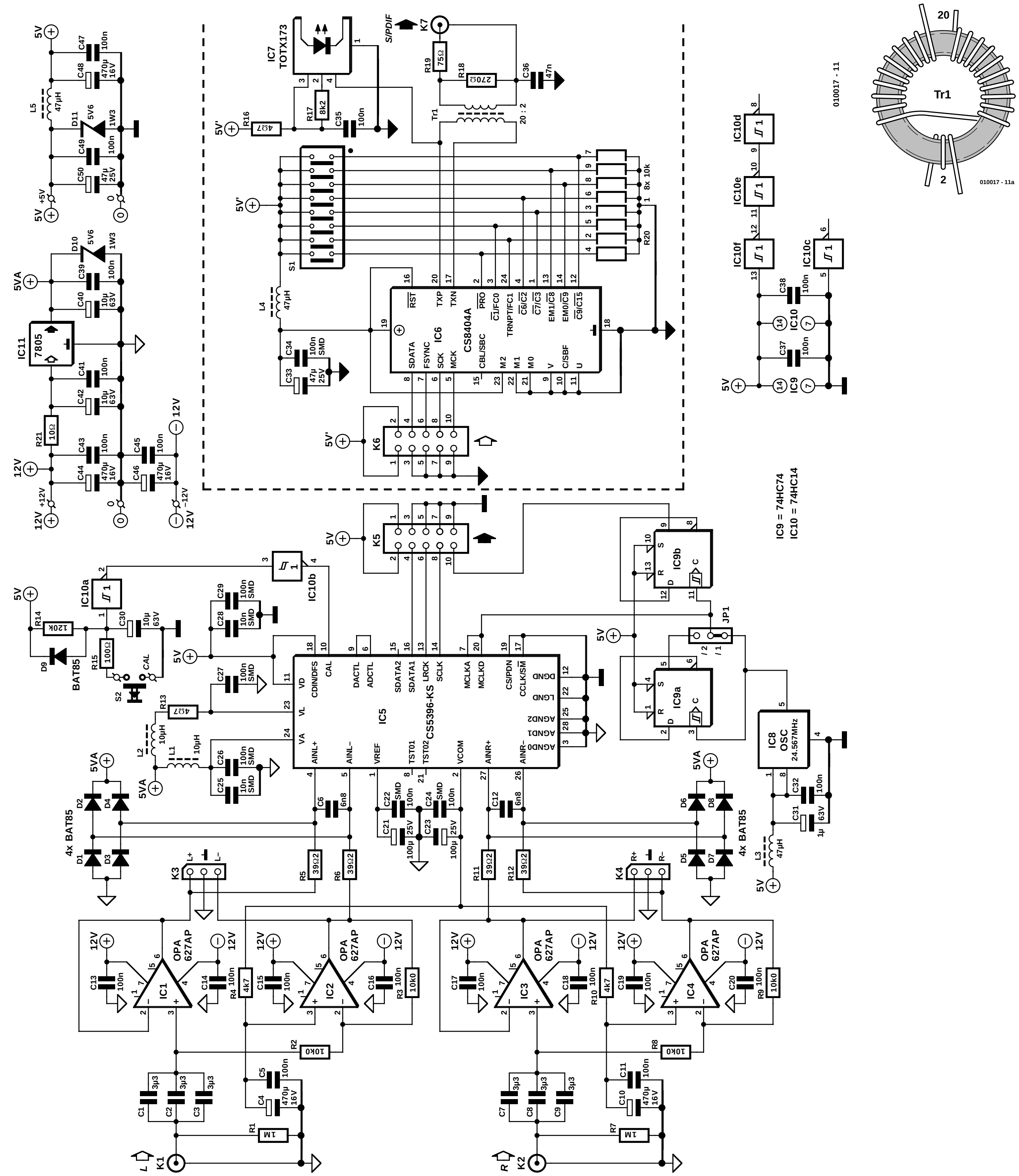


Figure 1. The actual converter consists primarily of a single IC, the CS5396-KS. The well-known CS8404A is used for the S/PDIF output.

for each channel. In order to allow asymmetric signals to be connected, AINL– and AINR– are provided with input signals via two inverters (IC2 and IC4). A pleasant side effect of this is that the input is thus made somewhat more sensitive (700 mV$_{rms}$ full scale).

Each of the input capacitors is made up of three 3.3µF MKT capacitors connected in parallel. They combine with the 10-kΩ input impedances (R2 and R8) to form high-pass filters with a corner frequency of 1.6 Hz. In order to allow an (external) symmetrical input stage to be connected, a

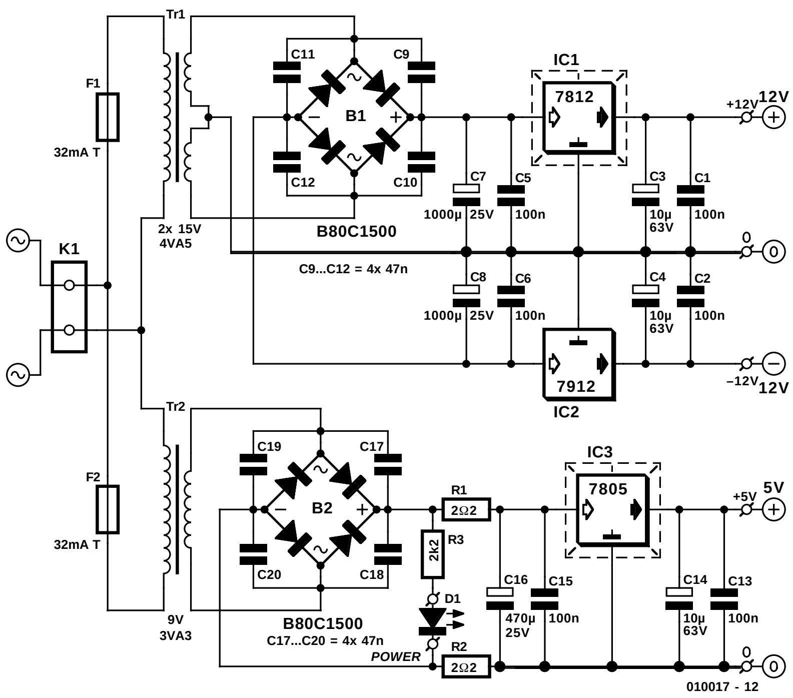


Figure 2. A double power supply is provided: one for the analogue part and one for the digital part.

three-way pin header is provided on the circuit board for each channel (K3 and K4). If this option is used, the input stages (IC1–IC4 and the associated components) should not be fitted. The pin headers can also be used as test points.

Converter

As shown in the schematic diagram, the A/D converter is liberally provided with decoupling. SMDs are used in part for this purpose, for the supply voltage as well as for the reference voltage and VCOM. Incidentally, the CS5396 is also an SMD (28-pin SOIC).

The converter is used in the stand-alone mode, with the sampling rate determined by the master clock. At 96 kHz, 64 times oversampling must be used, so a master clock frequency of 24.576 MHz is required.

Power Down (PDN) is not used, so this pin is tied to ground. DFS determines the output format; a High level here selects the I²S-compatible mode. S/M is Low to select Master mode and thus ensure that the digital outputs are all derived from the master clock.

The tri-level delta–sigma modulator is calibrated when power is first applied (CAL). The calibration sequence can be re-initiated during operation via S2 and IC10. During the measurements, it was found that the quality is improved if the converter is recalibrated after having warmed up for at least 30 minutes (do this with no input signal present).

The two divide-by-two circuits (IC9) provide the converter with a bit more flexibility with regard to the sampling rate. If you use a 24.576-MHz oscillator (IC8), you can choose a sampling rate of 48 kHz (÷2) or 96 kHz (÷1) using JP1. Alternatively, you can insert a 12.288-MHz oscillator, in which case your only choice is 48 kHz (÷1).

Output section

Connector K5 acts as a universal I²S output. All sorts of digital signal processing equipment can be connected here, such as a volume control, tone control, equalizer and so on. This output is also suitable for connecting a recording level meter, such as the digital VU meter described in the April and May 1996 issues of *Elektor Electronics* (but note that the pinout is different here).

Naturally, coaxial and optical S/PDIF outputs are also provided. This subcircuit is it built around the CS8404A (IC6), and is actually located on its own part of the circuit board that can be separated from the main board and then connected to the converter board using K5 and K6. This part of the output section is marked with by a dotted outline on the schematic diagram. The split design has the advantage that the digital output can be placed precisely where it best fits, such as immediately adjacent to the analogue outputs at the rear of the enclosure.

With regard to design, the CS8484A is practically identical to the CS8402 that we used in the previous circuit. Consequently, we need not describe its operation any further, beyond referring to Tables 5 and 6 in the manual for the CS5396 Evaluation Board (www.crystal.com) for the DIP switch settings. The default settings for dipswitch S1 are as follows:

S1-1	1
S1-2	0
S1-3	0
S1-4	0
S1-5	1
S1-6	0
S1-7	1
S1-8	1

where 1= closed and 0 = open.

Power supply

The power supply section (see **Figure 2**) has been kept separate from the rest of the circuit. Its design strongly resembles that of the power supply for the 'old' A/D converter. Here again, it has been completely split into separate analogue and digital parts. A standard symmetric ±12 V supply is used for the opamps, with the main components for this part being Tr1, B1, IC1 and IC2. Power for the digital part is supplied by the lower portion of the circuit, which basically consists of Tr2, B2 and the 5-V regulator IC3.

The CS5396 requires three supply voltages: two for the analogue side and one for the digital side. The voltage for the digital side is taken directly from the digital supply. The analogue +5-V supply is derived from the +12-V supply using a separate voltage regulator (IC11) on the converter circuit board. Two 5.6-V Zener diodes (D10 and D11) are provided to protect both the analogue and the digital +5-V supplies. Although it would have also been possible to work on the basis of a single transformer, with the analogue supply voltage being obtained by half-wave rectification of one half of the secondary winding and the digital +5 V from the other half, we intentionally chose to use two transformers. This lets us avoid unnecessary power dissipation in IC3, and it also has the advantage that the analogue and digital voltages can be completely separated from each

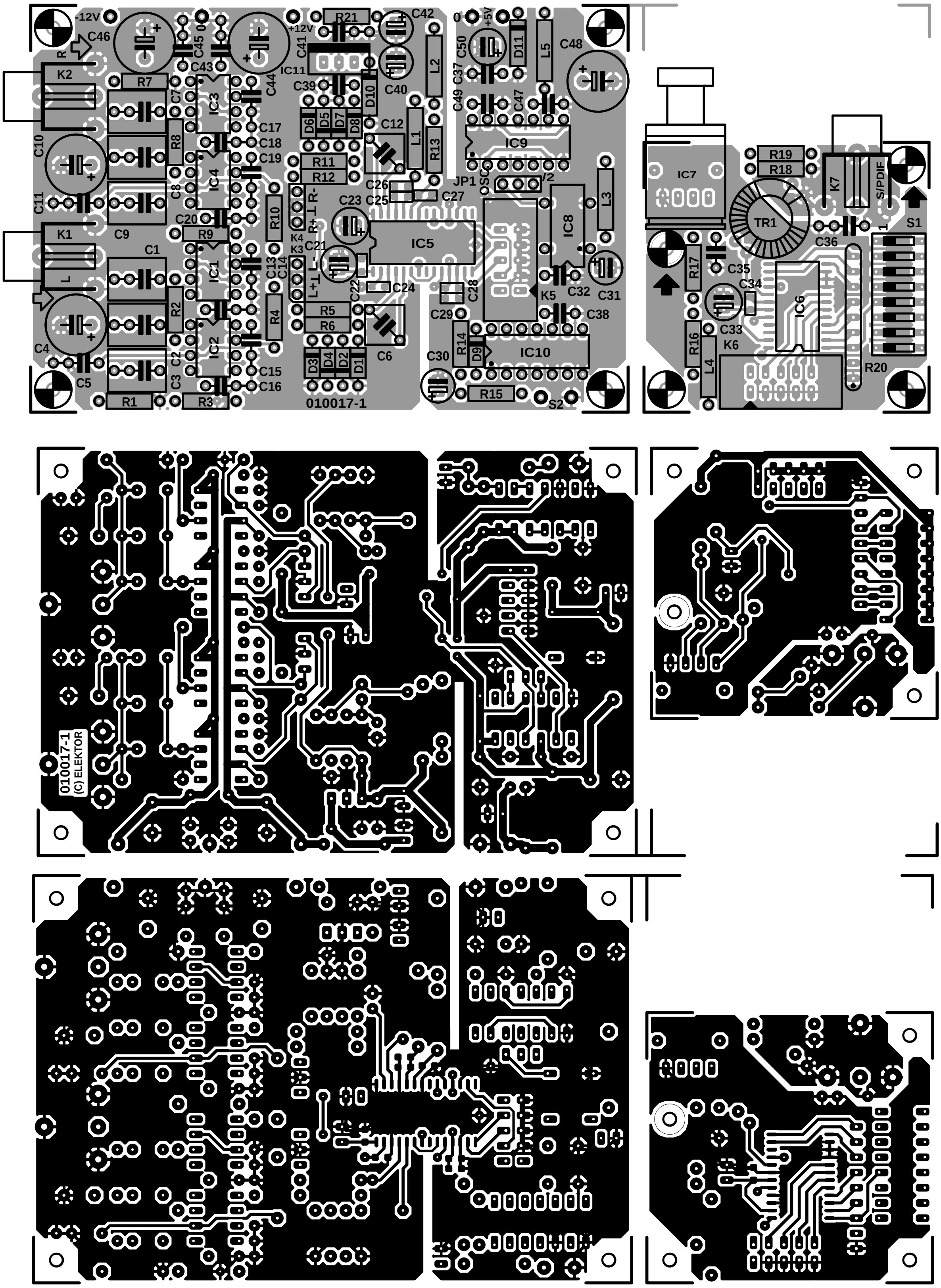


Figure 3. The copper track layout and component layout of the double-sided converter circuit board. The output circuit board can be sawn free from the rest of the circuit board.

other. Their ground points are tied together only on the converter circuit board next to the A/D converter.

The supply current of the CS5396, which is 285 mA in total, can be regarded as unusually high. The total dissipation of this small IC is more than 1.4 W, so it becomes quite hot! IC1 and IC3 on the power supply board also have to dissipate a fair amount of power, which is why heat sinks must be provided for these regulator ICs. For safety reasons, fuses are provided for the primaries of the two mains transformers.

Printed circuit boards

From the above, it should be obvious that there are two separate printed circuit boards: a double-sided board for the converter plus the S/PDIF output, and a normal single-sided board for the power supply. The copper track layouts and component layouts for these boards are shown in

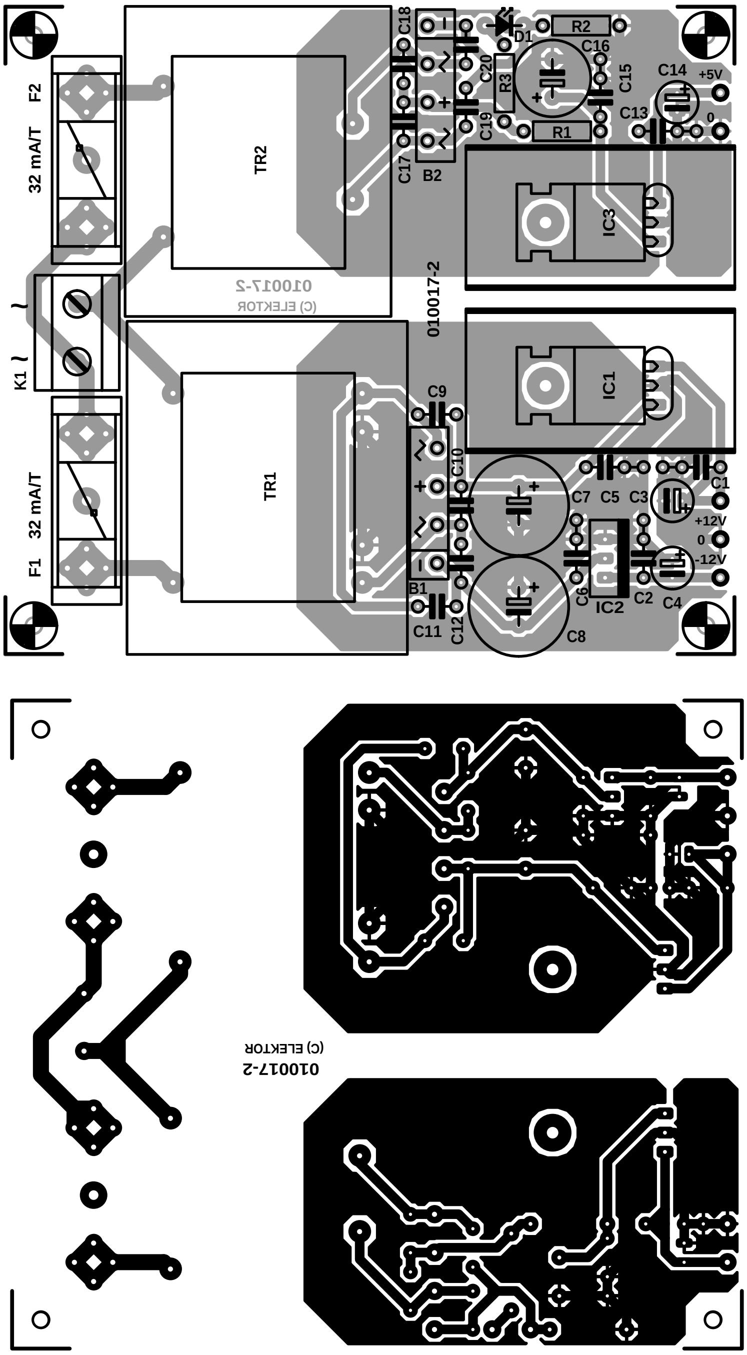


Figure 4. The power supply circuit board even has space for the two mains transformers.

COMPONENTS LIST
Converter board

Resistors:

R1,R7 = 1MΩ
R2,R3,R8,R9 = 10kΩ0
R4,R10 = 4kΩ7
R5,R6,R11,R12 = 39Ω2
R13,R16 = 4Ω7
R14 = 120kΩ
R15 = 100Ω
R17 = 8kΩ2
R18 = 270Ω
R19 = 75Ω
R20 = 10kΩ, 8-way SIL array
R21 = 10Ω

Capacitors:

C1,C2,C3,C7,C8,C9 = 3µF3 MKT, lead pitch 5 or 7.5mm
C4,C10,C44,C46,C48 = 470µF 16V radial
C5,C11,C13-C20,C32,C35,C37,C38,C39, C41,C43,C45,C47,C49 = 100nF ceramic
C6,C12 = 6nF8 1% polystyrene/polypropylene (EMZ)
C21,C23 = 100µF 25V radial
C22,C24,C26,C27,C29,C34 = 100nF SMD (case size 0805)
C25,C28 = 10nF SMD (case size 0805)
C30,C40,C42 = 10µF 63V radial
C31 = 1µF 63V radial
C33,C50 = 47µF 25V radial
C36 = 47nF MKT

Inductors:

L1,L2 = 10µH
L3,L4,L5 = 47µH

Semiconductors:

D1-D9 = BAT85
D10,D11 = 5V6 1W3 zener diode
IC1...IC4 = OPA627AP (Burr-Brown = Texas Instruments)
IC5 = CS5396-KS (Crystal) (Atlantik Elektronik)
IC6 = CS8404A (Crystal) (Atlantik Elektronik)
IC7 = TOTX173 Toshiba (Conrad Electronics)
IC8 = 24.576MHz oscillator block, type SG531P (Seiko Epson)
IC9 = 74HC74
IC10 = 74HC14
IC11 = 7805

Miscellaneous:

JP1 = 3-way pinheader + jumper
K1,K2,K7 = T-709G (Monacor/Monarch)
K3,K4 = 3-way pinheader
K5,K6 = 10-way boxheader, male (straight)
S1 = 8-way DIP switch
S2 = pushbutton, 1 make contact
Tr1 = ferrite ring core Philips TN13/7,5/5-3E25, primary: 20 turns, secondary: 2 turns 0,5 mm dia. (26SWG) enamelled copper wire

COMPONENTS LIST
Power Supply Board

Resistors:
R1,R2 = 2Ω2
R3 = 2kΩ2

Capacitors:
C1,C2,C5,C6,C13,C15 = 100nF ceramic
C3,C4,C14 = 10µF 63V radial
C7,C8 = 1000µF 25V radial
C9-C12,C17-C20 = 47nF ceramic
C16 = 470µF 25V radial

Semiconductors:
D1 = high-efficiency-LED, red
IC1 = 7812 + heatsink type ICK35SA (Fischer, Dau Components)
IC2 = 7912
IC3 = 7805 heatsink type ICK35SA (Fischer, Dau Components)

Miscellaneous:
K1 = 2-way PCB terminal block, lead pitch 7.5 mm
B1,B2 = B80C1500 (rectangular case) (80V piv, 1.5A peak)
Tr1 = 2 x 15 V/4VA5, e.g., VTR4215 (Monacor/Monarch)
Tr2 = 1 x 9 V/3VA3, e.g., VTR3109 (Monacor/Monarch)
F1,F2 = fuse 32mA/T (time lag)+ PCB holder

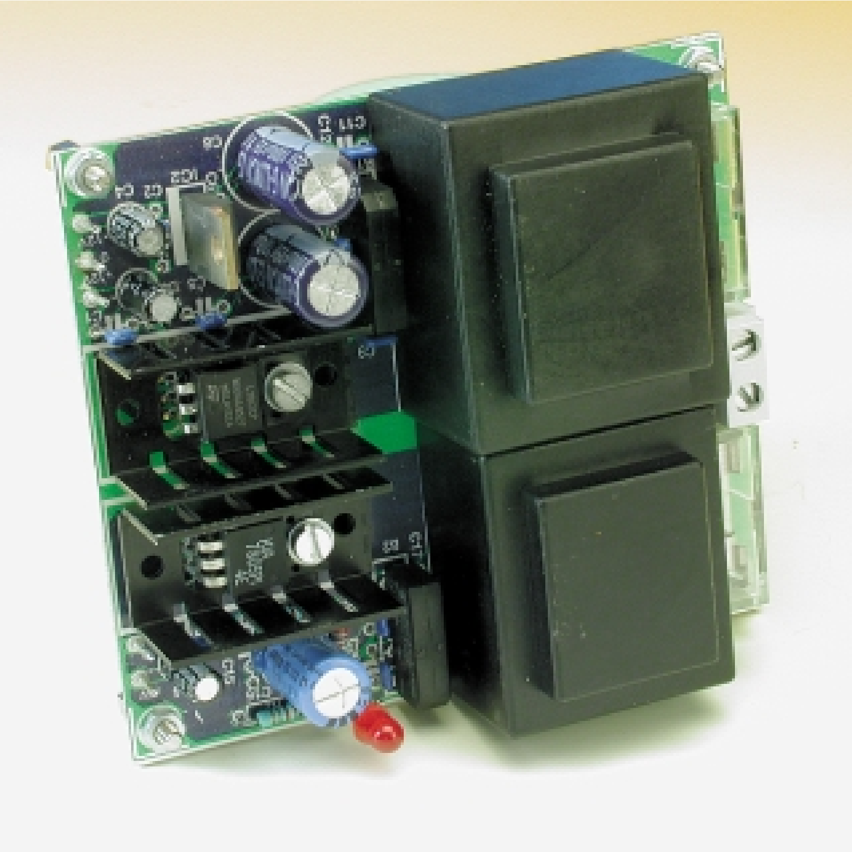


Figure 5. Small heat sinks must be provided for voltage regulators IC1 and IC3.

Figure 3 and **Figure 4**, respectively.

The construction of the power supply board is by far the easiest, since the dimensions of this board are quite generous and it holds relatively few components. If the transformers noted in the Components List are used, they can be mounted directly on the circuit board. IC1 and IC3 each have a small heat sink. The ±12-V and +5-V connections (and both ground connections) should be linked directly to the correspondingly marked points on the converter circuit board. The latter points can be found in the vicinities of C44/C46 and C50. The completely assembled power supply circuit board is shown in **Figure 5**.

The component density on the converter board is quite a bit higher, which means that the construction of this board requires significantly more care and attention. However, it shouldn't be all that difficult if you work methodically and stick precisely to the component layout and the suggested components. Capacitors C22, C24-C29, C34 and C35, as well as IC5, are SMD components, and soldering them requires a certain amount of experience. Soldering the capacitors is relatively easy if you first place them in the proper locations on the circuit board and hold them in place while soldering one end. You can then check their positions before soldering the other end. A soldering iron with a very fine tip is naturally essential, and it is important to pre-tin the connecting surfaces of the capacitors.

With the ICs, the procedure is roughly as follows: place the IC in its proper position on the circuit board and press it firmly against the board, solder a single pin at one of the corners, check the position and alignment of the IC, and then solder one pin at the diagonally opposite corner. After this, you can calmly solder the remaining pins with a steady hand. Make sure that no short circuits occur due to excessive solder; use solder braid to remove any superfluous solder!

As already noted, the S/PDIF output portion of the circuit board can be separated from the rest of the circuit board, and it is naturally best to do so before starting to assemble the board. A length of 10-way cable between K5 and K6 provides the connections between the two parts of the circuit board. The supply voltage for the output circuit board is also provided by this link.

The output transformer (Tr1) cannot be bought ready-made, so you will have to wind it yourself. That sounds more difficult than it actually is. The transformer consists of a 20-turn primary and a 2-turn secondary on a TN13/7,5/5-3E25 toroidal core. Use enamelled copper wire with a diameter of 0.5 mm. First wind the 20 turns for the primary and distribute them neatly over the entire core. Leave a bit of space in the middle, where you can then wind the two turns of the secondary winding. That's all there is to it!

The fully assembled converter board is shown in **Figure 6**, and the small output circuit board is shown assembled in **Figure 7**. The output transformer can be clearly recognised in the latter photo.

Before putting the converter into service, it is a good idea to once again carefully check the assembled circuit boards with reference to the photos of the assembled boards and the component layouts shown in **Figure 3** and **Figure 4**. After this, it is a

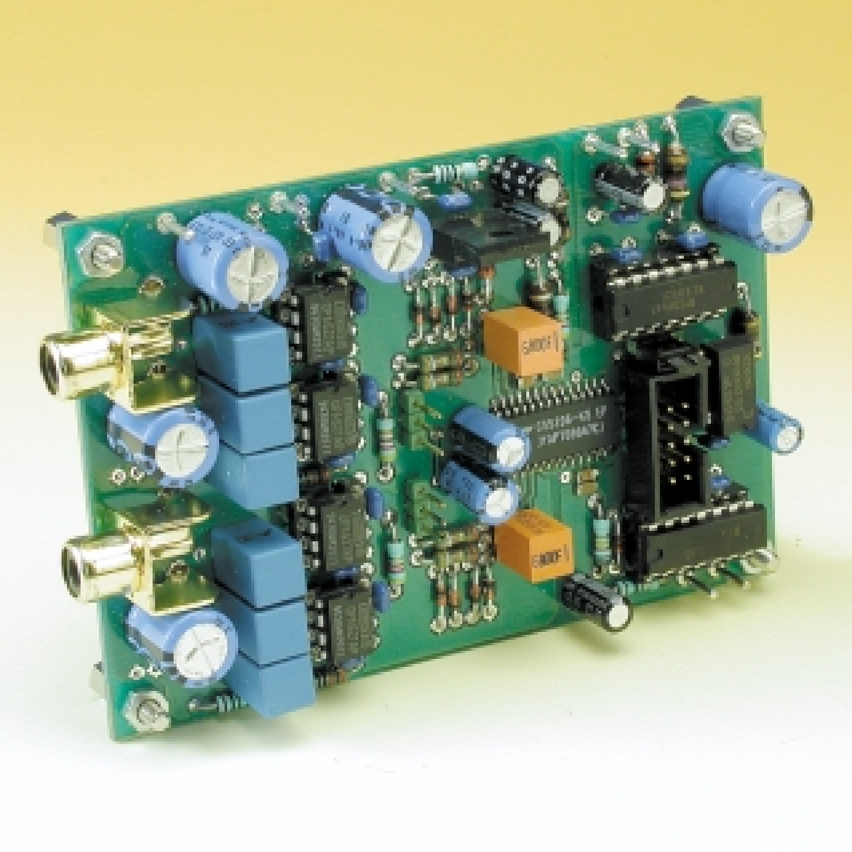


Figure 6. The fully assembled converter board.

good idea to check whether the power supply provides the correct output voltages. To do so, temporarily connect 2.2-kΩ resistors between terminal pairs PC1–PC2, PC3–PC2 and PC4–PC5, and connect K1 to the mains using a well-insulated mains cable (the on/off indicator D1 should be illuminated). Now you can check the three output voltages using a multimeter. Only after you are sure that everything is in order should you remove the resistors and connect the ±12 V and +5 V terminals to the converter board.

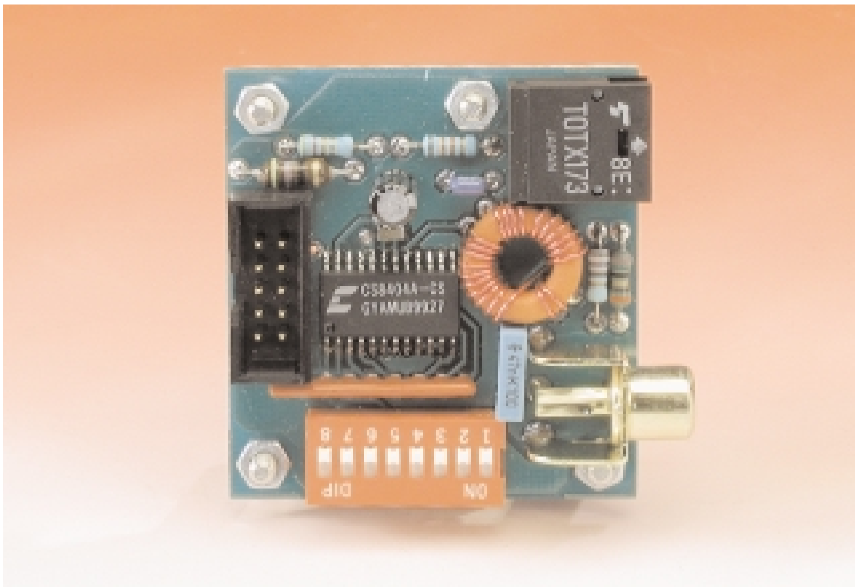


Figure 7. The fully assembled output circuit board. The toroidal transformer can be clearly seen.

Enclosure

Depending on the application, this A/D converter can be built into an existing piece of equipment or provided with its own enclosure. In the latter case, we would expressly advise you to use a metallic enclosure.

If the circuit boards are mounted such that the input and output connectors (K1, K2 and IC7, K7, respectively) project through the rear panel of the enclosure, the necessary wiring can be kept to a minimum. In this case, only the power supply has to be connected to the converter board, and the link to the output board must be provided.

Naturally, a mains power input socket must be mounted at the rear of the enclosure, and this must be connected to K1 on the power supply board via a length of well-insulated mains cable. A mains power switch can be placed between mains input and K1, if so desired. The mains switch, on/off indicator D1 and calibration pushbutton S2 are the only front-panel controls and indicators. Pay careful attention to electrical safety with regard to mounting the mains input, cable and switch, and try to make the most of the advice given in the Safety Guidelines published from time to time in *Elektor Electronics* (e.g. January 2000).

For a metallic enclosure to provide effective screening, it is essential that it be tied to circuit ground. This can be easily done by connecting a wire between the ground terminal of input socket K1 or K2 and a solder lug fastened to the enclosure. Finally, place a label on the back of the enclosure that indicates the mains supply voltage, the fuse value and the circuit number.

Conclusion

We probably don't have to say very much about using the A/D converter. Its most obvious application is as a supplement to digital audio equipment whose A/D conversion is no longer up to date. In this regard, we have a handy tip for audio fans. Terratec can presently supply sound cards that support 24-bit, 96-kHz recording, in a wide variety of price classes. The converter described here would certainly be a major improvement to the converter that is already present on the sound card, especially since this converter is located external to the PC and can be connected to the sound card via an electrically isolated TosLink interface.

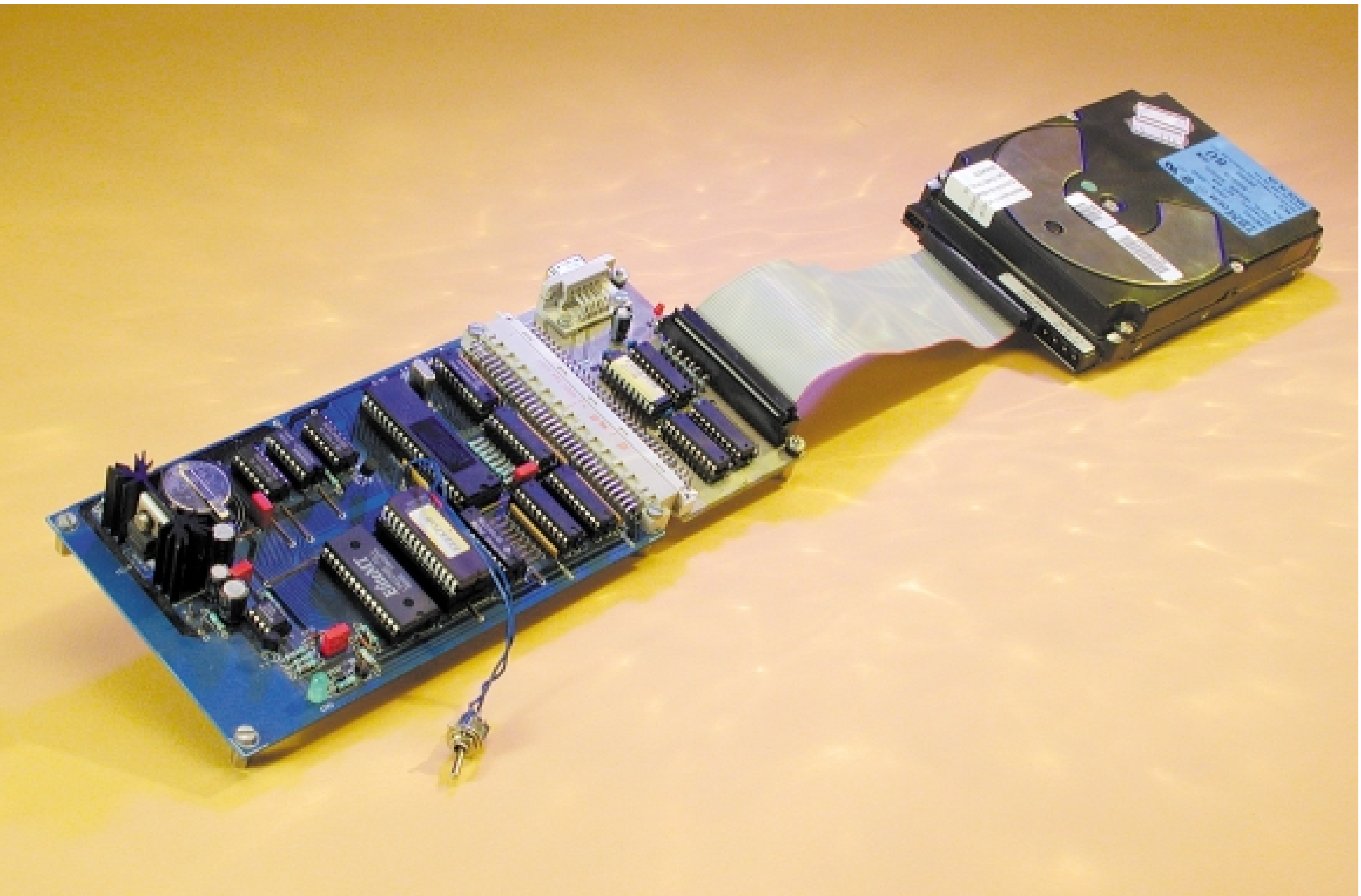
(010017-1)

IDE Hard Disk Interface for 8-bit Controllers

controller seeks attachment with a hard disk…

Design by S. Schwark

Although 8-bit operation is included in the IDE interface, most hard disk drives do not support this mode. If you nonetheless want to control such drives with an 8-bit microcontroller, you will find the interface circuit described here to be essential.



Hard disk drives as mass storage devices are an indispensable part of modern computers. A few years ago, they were large and expensive devices, but presently (and up until the next generation) they are available with enormous capacities and small physical dimensions at quite reasonable prices. Most contemporary computers use hard disk drives with IDE interfaces. Since this interface (which is also referred to as ATA) comes from the PC world and thus works with a 16-bit operating system, it is possible to use it directly with 8-bit controllers, albeit with some problems. Although the interface can be switched to the 8-bit operating mode, this does not always work in practice, since many hard disk drives do not support this option. We are thus forced to provide a 16-bit data bus in the microcontroller circuit. This is precisely the task of the interface circuit described here. It has been developed to allow hard disk drives to be controlled by the 8032 Single Biard Computer, which was originally described in articles published in *Elektor Electronics* in 1991. Due to its relatively uncomplicated design, it can also be used relatively easily with other microprocessors.

The Integrated Disk Electronic Interface (IDE) was developed in 1984 by Compaq. The objective was to integrate the hard disk controller (which up to then had been a rather large expansion card) into the disk drive. The PC expansion card could then be made much simpler, since all that was needed was a connection to the system bus. The IDE interface is in principle nothing more than an AT slot, reduced to the leads required by the hard disk drive. This is why this interface can also be properly called ATA (AT Attachment).

A PC communicates with a hard disk drive via addresses 1F0h–1F7h and 3F0h–3F7h. There are also two CS (Card Select) leads, which select individual regions in the hard disk drive. The actual register selection takes place using three address lines. In the interface described here, different addresses are used. Both register banks lie in the address range C000h–C00Fh. The individual registers and their functions are described below under ‘Software’.

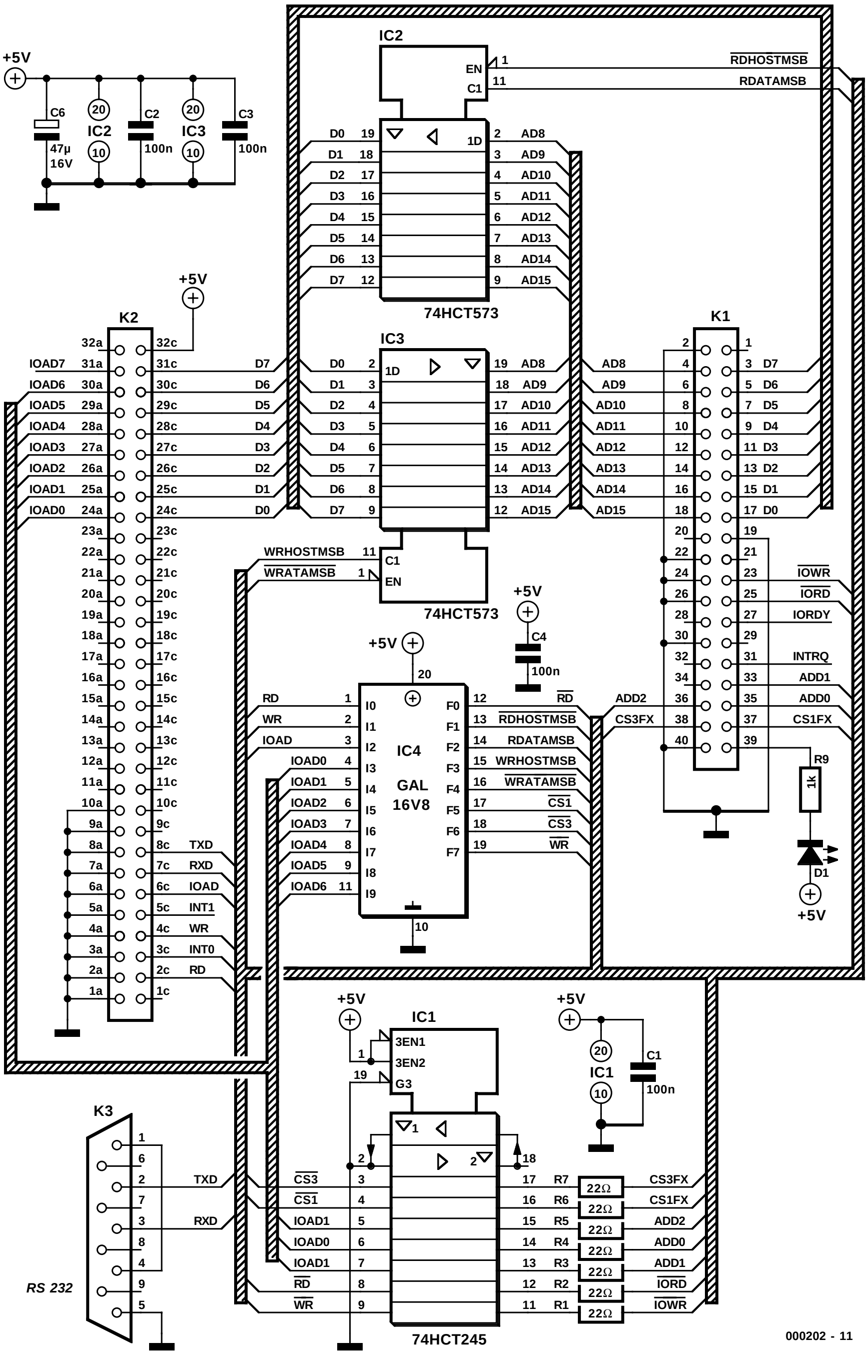


Figure 1. A GAL IC forms the heart of the interface circuit.

The ATA command set is used to communicate with the hard disk. This defines the codes that initiate the various functions (see **Table 1**). The commands for the hard disk are written to the command register at address 1F7h (corresponding to C00Fh). The individual commands are defined in the ATA command set specification. However, there are also vendor-specific variations, so some drives do not recognise some of the commands.

Since its original development, the ATA interface has achieved a considerable range of distribution, and today it is used for hard disks and CD-ROM drives in most PCs. Since the original specification was not suitable for devices such as CD-ROM drives, an extension was introduced a few years ago. The extended interface, which is called ATA Packet Interface (ATAPI), fully matches the ATA interface as far as the hardware is con-

COMPONENTS LIST

Resistors:
R1-R7 = 22Ω
R8 = 1kΩ

Capacitors:
C1-C4 = 100nF
C5 = 47µF 16V

Semiconductors:
D1 = high-efficiency LED, red
IC1 = 74HCT245
IC2,IC3 = 74HCT573
IC4 = GAL 16V8 (programmed, order code **000202-31**)

Miscellaneous:
K1 = 40-way boxheader with polarisation
K2 = 64-way DIN41612-C socket (female), angled pins, PCB mount
PCB, order code **000202-1**
Disk, order code **000202-11** (example program in C and GAL listing)

cerned. The transfer modes and registers are the same. While most normal ATA commands consist of only a command code that is written to the command register, ATAPI employs a 12-byte command packet. This structure of this data packet is strongly based on the SCSI interface, so it is understandable that IDE CD-ROM drives and streaming tape drives (ZIP) are actually SCSI devices that conduct their communications via a sort of translation step.

Hardware

Four ICs suffice for the hardware of the interface circuit. The heart of the circuit, as shown in **Figure 1**, is a GAL16V8 (IC4) that is used to drive latches IC2 and IC3. The source code for programming the GAL can be downloaded free of charge from the Free Downloads section on the *Elektor Electronics* Internet site, and it is also available on diskette from Readers Services (number **000202-11**). The GAL generates some of the control signals needed for the ATA interface, such as the two card select signals CS1Fx and CS3Fx. These two signals select the two register banks that are available in the hard disk drive. The designations of the two leads come from the PC world, where these registers are located at addresses 1F0h–1F7h and 3F0h–3F7h, respectively. The GAL of the interface circuit uses the address region C000h–C00Fh, into which all the registers are mapped. Individual registers are selected using address lines A0–A2. The individual registers are listed below under 'Software'.

The GAL also generates the two signals /WR and /RD by inverting the controller signals RD and WR. The remaining outputs control IC2 and IC3. IC2 is used for temporary storage of the MSB when the processor reads a data word from the bus. Whenever the host sends a read command via the ATA interface, IC2 loads the MSB from data lines D8–D15 into its memory. If the processor then reads address C000h, IC2 passes the most recently read

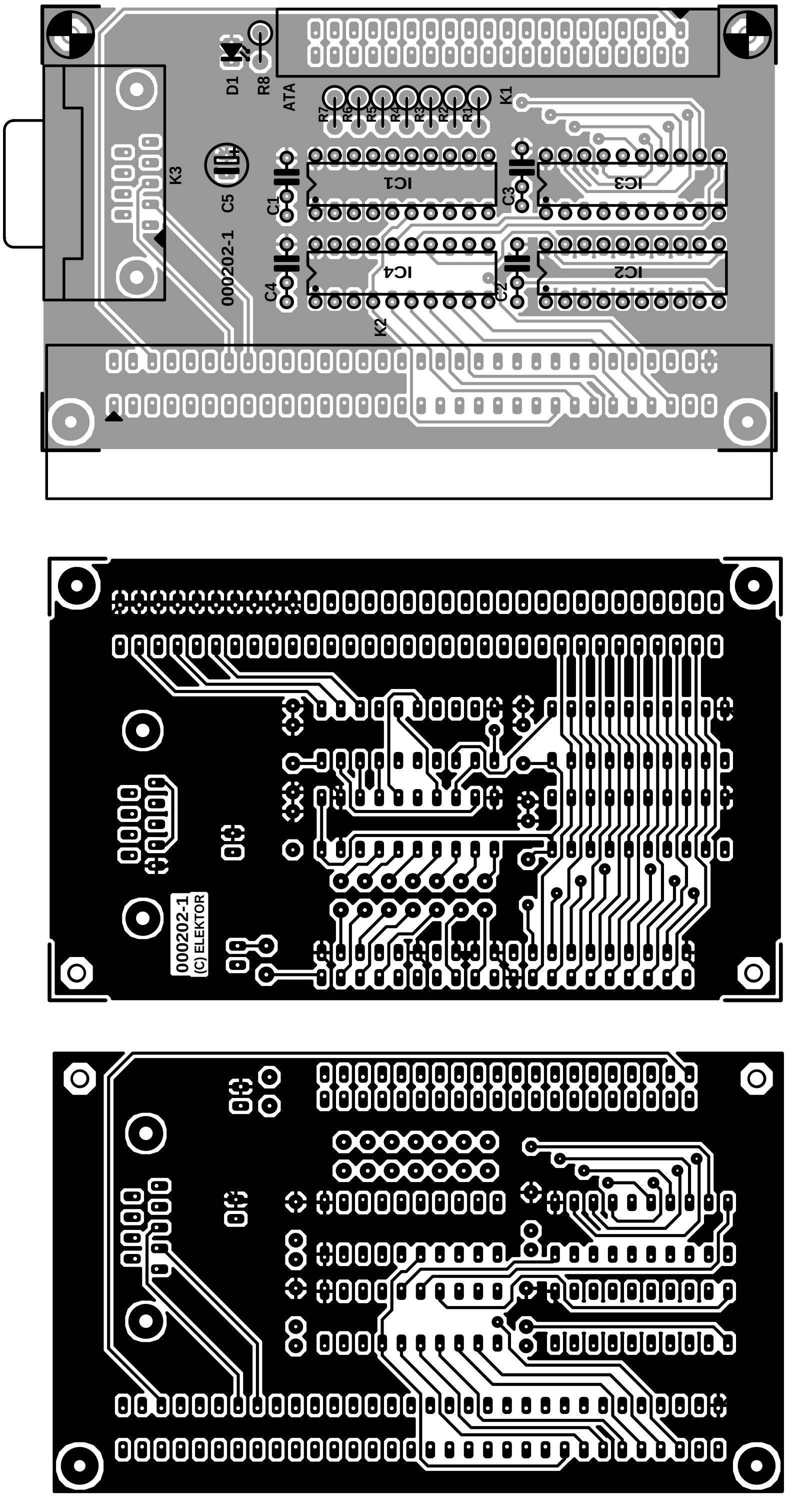


Figure 2. The track layout and component layout for the small interface circuit board.

Table 1. ATA commands

Command	Code	FR	SC	SN	CY	DH
CHECK POWER MODE	E5h		x			D
EXECUTE DEVICE DIAGNOSTIC	90h					D
IDENTIFY DEVICE	Ech					D
IDLE	E3h		x			D
IDLE IMMEDIATE	E1h					D
INITIALIZE DEVICE PARAMETERS	91h		x			x
READ MULTIPLE	C4h		x	x	x	x
READ SECTOR(S)	20h 21h		x	x	x	x
READ VERIFY SECTOR(S)	40h 41h		x	x	x	x
SEEK	70h			x	x	x
SET FEATURES	Efh	x				D
SET MULTIPLE MODE	C6h		x			D
SLEEP	E6h					D
STANDBY	E2h		x			D
STANDBY IMMEDIATE	E0h					D
WRITE DMA	CAh CBh		x	x	x	x
WRITE MULTIPLE	C5h		x	x	x	x
WRITE SECTOR(S)	30h 31h		x	x	x	x

SC = Sector Counter C00Ah
SN = Sector Number register C00Bh
CY = Cylinder registers C00Ch and C00Dh
DH = Head Number register C00Eh
Note: 'D' means that only the device parameters in the register are significant.

MSB to the processor via the data bus. IC3 performs the same task when a write command is issued via the ATA interface. The host writes the relevant MSB to address C000h, which causes IC3 to transfer this byte into its memory. If the hard disk wants to read the data, the IC puts the MSB on data lines D8–D15. The final IC, IC1, is used to buffer the control signals that link IC4 and the processor to the ATA interface.

In addition to the 40-pin ATA connector (K1) and the 64-pin DIN4162-C connector for the link to the microcontroller board, there is a 9-pin Sub-D connector on the circuit board. The serial lines of the controller board are looped through this connector. Capacitors C1–C5 are the usual decoupling capacitors, and LED D1 indicates disk activity (the drive pulls this lead to ground when the disk is active). No distinction between master and slave drives is possible, since both drive signals use the same pin.

Table 2. Partition table entries

Offset address	Contents
00h	partition status (00h = inactive, 80h = boot partition)
01h	read/write head for the start of the partition
02h	sector and cylinder for the start of the partition
04h	partition type (e.g. 04h for DOS 16-bit FAT)
05h	read/write head for the end of the partition
06h	sector and cylinder for the end of the partition
08h	offset from the partition sector to the first sector of the partition (boot sector), in sectors
0Ch	number of sectors in this partition

Assembling the interface circuit board (**Figure 2**) should not present any problems. Sockets may be provided for the ICs. When soldering in the components, pay attention to the polarity of C6 and the LED, and naturally also to the orientation of the ICs.

Software

In order to drive the ATA interface directly from a microcontroller, you naturally need not only the necessary hardware, but also suitable software for writing the individual commands to the disk drive registers, which are described below. The Compuboard communicates with the ATA interface via the addresses that are listed after the register names. Some of the registers have different meanings if the disk drive is operated in LBA (Logical Block Addressing) mode. In this case, instead of addressing the sectors using cylinder, head and sector numbers, the individual disk sectors are numbered in a linear sequence. LBA was introduced to eliminate the ATA interface's capacity limit of 504 MB (you may also see 528 MB, but this is calculated using 1000 bytes for a kilobyte). The first sector has an address of '0'. With a 504-MB disk, the address of the final physical sector is '1,065,456'. Using this type of addressing, the capacity limit can be extended to 7.8 GB. If you want to use a hard disk that has a capacity of more than 504 GB, you thus have to operate the drive in LBA mode.

Data register C000h

Data bits 8–15 are read and written via this register.

Data register C008h

Data bits 0–7 are transferred via this register. The data are valid when the DRQ bit is set in the Status register.

Error Bits / Feature C009h

With a read access, the error bits are returned in this register. These bits can be used to determine the nature of any error that has occurred. With a write access, parameters are passed via this register (for example, cache on/off).

Sector Counter C00Ah

This register contains the number of sectors to be processed in the following drive access. Its meaning is the same for write and read accesses.

Start Sector C00Bh

The start sector for the subsequent read or write access can be written to this register. In LBA mode, LBA bits 0–7 are located here.

Cylinder LSB C00Ch

The lower eight bits of the cylinder address are located here. In LBA mode, this register contains LBA bits 8–15.

Cylinder MSB C00Dh

This contains either the upper eight bits of the cylinder address or LBA bits 16–23.

Drive / Head Number C00Eh

Bits 0–3 contain the binary-coded head number. In Normal mode, there can be up to 16 heads. Bit 4 distinguishes between the Master and Slave drives (1 = Slave, 0 = Master). Bits 5–7 contains the number of bytes per sector. A fixed value of 512 bytes/sector is entered here. In LBA mode, LBA bits 24–27 are located here.

Status / Command register C00Fh

The ATA commands are written to this register. For read accesses, this register is the Status register. The current status of the drive can be recognised from the individual bits:

Bit 0 (ERR): indicates an error. The nature of the error can be determined from the error bits at address 1F1h.
Bit 1 (IDX): index pulse (vendor-specific).
Bit 2 (CORR): indicates that the data being transferred have been corrected using ECC.
Bit 3 (DRQ): the device is ready for data transfer.
Bit 4 (DSC): the search command has been executed.
Bit 5 (DF): a write error has been detected.
Bit 6 (DRDY): drive ready (following switch-on, for example).
Bit 7 (BUSY): the drive is busy executing a command.

2nd Status register / Device Control C006h

A read access here yields the same result as an access to the first Status register, except that the state of the interrupt line is not changed. The interrupt can be activated by setting bit 1 to ‘0’ with a write access. A software reset of the disk drive can be initiated by setting bit 2 to ‘1’. Bits 3–7 are not used, and bit 0 must always be ‘0’.

Active Address C007h

This is a read-only register. Bits 0 and 1 distinguish between Master and Slave drives. The ones-complement value of the enabled head is located in bits 2–5. Drive write activity is indicated by a ‘0’ in bit 6. Bit 7 indicates to the PC that the diskette has been removed from the floppy disk drive. This bit is not used for our purposes.

Table 3. ATA interface connector pin assignments (K1)

Pin	Signal	Description
2,19,22,24, 26,30,40	GND	ground
1	RES	reset signal from the computer
17	D0	D0–D12 form a bi-directional data bus between the hard disk drive and the computer. The data, status and control information are transferred via this bus. If the drive is not selected, these leads are switched to a high-impedance state.
15	D1	
13	D2	
11	D3	
9	D4	
7	D5	
5	D6	
3	D7	
4	D8	
6	D9	
8	D10	
10	D11	
12	D12	
14	D13	
16	D14	
18	D15	
20	Keying pin, cut off to provide protection against connecting the cable the wrong way around.	
21	/IOCHRDY	The computer must wait until this signal is inactive before accessing the drive.
23	/IOWR	write signal for I/O port addresses
25	/IORD	read signal for I/O port addresses
27	/IOCHRDY	same as pin 21
28	ALE	Address Latch Enable
29	frei	reserved for future extensions
31	IRQ14	interrupt request to the computer
32	IO16	indicator for 16-bit data transfers
34	/PDIAG	Passed Diagnostics: the Slave drive reports to the Master that the internal diagnostics have been completed.
33	A1	A0–A2 are address lines from the computer, used to select the individual registers
35	A0	
36	A2	
37	/CS0	addresses 1F0h through 1F7h
38	/CS1	addresses 3F0h through 3F7h
39	/ACT	Can be used to drive an LED to indicate drive activity.

In order to make programming somewhat easier, the author has written a small C program that demonstrates how a drive can be controlled. This program is also available via Internet and on the diskette. The program ATA.C first initialises the serial interface, since later on messages will have to be output via this interface. Next, an interface reset is triggered. If the drive then reports itself to be ready, the program sends an IDENTIFY DEVICE command to the hard disk drive. The hard disk then starts to output its internal data. The individual data items specify the model, version, serial number, number of heads and cylinders and so on. The data sent by the hard disk drive are output via the serial interface. Next, the first data block of the boot sector is read and also output. Here it is important to always read out the full 512 bytes, since otherwise problems can occur later on with read processes. This small example program provides a good indication of how easy it is to communicate with an IDE hard disk drive. There are also several MCS-51 assembler routines available, which you can incorporate into your own programs.

Hard disk organisation

In order to be able to find data that you have written to a hard disk, you naturally need some sort of structure. The first level of the hard disk structure is called ‘low level formatting’. This consists of defining tracks on the surface of the disk, with multiple sectors per track. Each sector can hold 512 bytes. Following low level formatting, the hard disk is like an enormous diskette. This state would be adequate for a few applications. For example, if you only want to store data for an extended period, you could just write each sector in turn with 512-byte data packets. Normally, however, this sort of sequential utilisation of the disk is not a satisfactory solution.

A normal PC user does not have to be concerned with the exact operation of the hard disk drive. This task is looked after by the device drivers of the operating system or the BIOS. However, if you want to be able to access a disk using software that you have written yourself, there’s no getting around the internal structure of the disk.

A hard disk contains one or more partitions. These partitions, which are also called ‘volumes’, are distinct, precisely demarcated regions of the hard disk. Their purpose is to make available different hard disk regions the can be managed by different operating systems.

After executing the memory test, running the test for various basic functions of the motherboard and initialising the graphics card, the BIOS loads the first sector of the hard disk drive (head 0, cylinder 0, sector 1), which is the partition sector. If the BIOS finds the code sequence ‘55AAh’ in the final two bytes of the 512 sector bytes, it recognises the sector as executable code and starts program execution at address 0000:7000 with the first byte of the sector. The job of the code stored at this address is to detect the active partition and to execute the contents of the boot sector available in that partition. The routine can determine where to find this sector from the contents of the partition table (which normally starts at offset 01BEh in the partition sector). This table contains a 16-byte entry for each partition. The routine can determine which partition is the active boot partition from the first byte of the table entry, which is ‘80h’ for an active partition and ‘00h’ for an inactive partition. If no partition is active, or if more than one partition is active, the routine terminates execution.

The meaning of the partition table entries is listed in **Table 2**. Although these entries are not mandatory, they are present for practically every operating system. Of course, the operating system cannot do very much with the partition alone. This region must first be formatted. This amounts to arranging the physical sectors into logical regions (clusters), which the operating system in question can use coherently. The administration of these clusters depends on the operating system that is used. DOS uses the File Allocation Table (FAT). Each entry in this table designates one of these clusters. It contains information such as which file this cluster belongs to and which cluster contains the following portion of the file. You will have refer to the documentation for the operating system in question to find out how the FAT is used for reading and writing files, since it is not possible to deal with this subject within the confines of this article.

The content of byte 04h indicates the following to the operating system:

Byte 04h	Operating System
00h	unused partition
01h	DOS 12-bit FAT
04h	DOS 16-bit FAT
05h	extended partition
08h	OS/2
0Bh	FAT 32
41h	PowerPC boot
51h	Novell
64–69h	Novell
81h	Linux
83h	Linux swap
A5h	FreeBSD
A9h	NetBSD

Table 4. Memory structure of the ATA interface

Address	Read	Write
C000h	D8...D15	
C006h	alternate status	device control
C007h	drive address	
C008h	Data	
C009h	error	Feature
C00Ah	sector count	
C00Bh	sector number	
C00Ch	cylinder low	
C00Dh	cylinder high	
C00Eh	drive/head	
C00Fh	staus	Command
Status register		
D7	BUSY	device busy
D6	DRDY	device ready
D5	DF	devive fault
D4	DSC	device seek complete
D3	DRQ	data request
D2	CORR	correction
D1	IDX	vendor specific
D0	ERR	error
Device control register		
D7...D23		reseved
D2	SPST	soft reset
D1	NIEN	interrupt enable (activelow)
D0	0	always 0

The capabilities of this interface circuit extend beyond just connecting a hard disk drive. Since the ATAPI CD-ROM interface employs the same register model, the circuit presented here can also be used for a CD-ROM drive. For these ATAPI devices, there is a series of special commands that can be used to activate the normal playback functions in a fairly straightforward manner. Maybe you can find a new use for that old 2x-speed CD-ROM in your junk box?

(000202-1)

You can find Internet addresses for information related to this subject at the Internet site of the T13 committee, which is responsible for the specification of the ATA standard:

http://www.t13.org

IR Remote Control Codes (1)

formats, protocols and (in)compatibility

By A.N. Other

There are so many different remote control message formats currently in use that it can all be a bit confusing. If you are experimenting with remote controller IC's then its important to know the different control protocols used by each manufacturer. This article seeks to describe the most popular protocols in current use.

Its almost impossible to buy a TV today that doesn't have a IR remote controller and its only when we temporarily mislay this device that we realise how useful they are. Back in 1975 when the first remote controllers appeared they used ultra sonic signals to send the control information, these were later superseded by the controllers that we are familiar with today using infra red. The infra red devices offer lower production costs, wide operating range and good communication security. A look inside a typical remote controller will show that it consists of only one IC. The IC interprets each key press and sends a coded data signal to the transmitter IR diode. A simple resonator is also used to supply a stable clock. On the receiving side in the equipment being controlled we find a IR detector and demodulator which is normally integrated into the same device. The SFH505-xx family of devices from Siemens contains the receiver, demodulator and output driver so that the received data can be connected directly to a micro controller or control decoder. Unfortunately the actual control protocol used by each manufacturer are mostly incompatible.

The IR message transmitted by the controller is subject to interference from other IR sources in the vicinity. These include heaters, incandescent lamps and other heat generators. One standard method of rejecting this unwanted interference is to modulate each transmitted bit with a stable carrier frequency in the range of 30 to 40 kHz. Another method is the so-called flash mode, this technique is employed by the Plessey MV500 chip (described in the February 1991 edition of *Elektor Electronics*). This method outputs data in the form of 17 µs short flashes of IR light followed by different off periods. Nokia also use this method with their IRT1250 IC. This system has however not gained wide acceptance and the vast majority of remote controllers use the modulation technique.

The accompanying oscilloscope pictures show each of the described transmission formats received by the Temic TFMS5360 receiver IC. This device is optimised for reception of a signal modulated at 36 KHz but can also detect other frequencies albeit with a reduced range. In the upper half of the picture a single telegram is shown and in the lower half a continuous key-press is shown. The output of the IC goes low when the modulated signal is detected.

It is important to note that the equipment manufacturer is entirely at liberty to choose a transmitter clock frequency and as such the timing given here may not be accurate under all conditions. The timings of pulse lengths may also be affected by the sample clock in the TFMS5360 and could have an error of ±160 µs (Temic data sheet).

The communications formats described are the most popular but it does not represent all the possible formats that you are likely to find. Many firms have devised their own control format, sometimes in order to reduce costs or sometimes to incorporate different control features that are not catered for with the existing standards. If you use a mask programmable micro controller for coding and decoding you will be completely at liberty to devise your own protocol which may be more suited to your own particular hardware of software. This method also ensures that a manufacturer will not need to worry about licencing fees or possible patent infringement.

Some modern remote IR controllers transmit the message a number of times using different message formats. For example the controller will first send out the Japanese code and then 50 ms later sends out the same command but this time using RC 5 code. The advantage here for the equipment manufacturer is that for future equipment development you need not wait for a chip manufacturer to produce a controller using a particular IR coding standard. It is now possible to select the best or cheapest integrated equipment controller and be sure that the IR remote controller will produce compatible control signals.

So which manufacturer and which coding system? This article describes some of the most popular IR coding standards currently in use.

(010023-1)

Code	Manufacturer
RECS80	Thomson, Nordmende
NEC	Harman/Kardon, Yamaha, Canon
DENON	Denon
SIRCS	Sony
RC5	Loewe, Philips, Grundig, Marantz
MOTOROLA	Grundig, Kathrein
JAPAN	Panasonic, Loewe
SAMSUNG	Samsung
DAEWOO	Daewoo

RC5 Code

The most widely used coding method for IR control in Europe is the RC 5 code. This was originally developed by Philips and has the capacity to send 2048 different commands. 32 addressable groups each with 64 commands. Each piece of equipment has its own address so that for example adjusting the volume of your audio system will not affect the sound level of your TV. One complete message has a length of 14 bits and is composed of the following bits:

– 2 Start bits to control the AGC levels (Auto gain control) in the receiver IC.
– 1 Toggle bit indicates that a new key is pressed.
– 5 System address bits
– 6 command bits

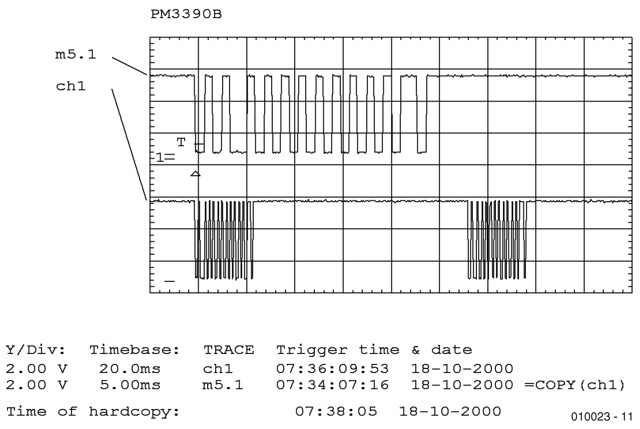
The toggle bit changes its value every time a new key is pressed and is used to tell the difference between pressing the key again and holding the key down. The five address bits follow the toggle bit and indicate which piece of equipment is being controlled. Lastly the six command bits contain the control information.

RC5 code employs biphase encoding, One bit of data is represented by two half bits. A Low/High combination of these bits indicates a data ‘1‘ whereas a High/Low combination indicates a data ‘0‘ The length of each bit is 1.778 ms, and a complete message is 24.889 ms long.

The RC5 code is probably the best documented protocol for IR control and particularly interesting are the two free system addresses 7 and 13 these are not allocated to any particular equipment type but are reserved for experimental purposes. Typical IC’s used for this message format are:

Transmitter:
SAA3006, SAA3010 (Philips)
HT6230 (Holtek)

Receiver:
SAA3009, SAA3049 (Philips)



```
Y/Div:   Timebase:  TRACE   Trigger time & date
2.00 V    20.0ms    ch1     07:36:09:53  18-10-2000
2.00 V    5.00ms    m5.1    07:34:07:16  18-10-2000 =COPY(ch1)
Time of hardcopy:             07:38:05  18-10-2000
```

010023 - 11

Figure 1. RC5 code at the output of the receiver IC TFMS5360.

Table 1 shows in decimal the correspondence between the equipment and command codes used for this format.

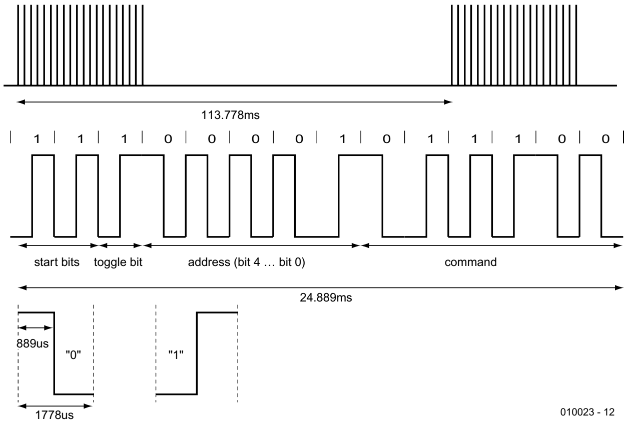


Figure 2. RC5 code message format (Address 1, command 28 shown).

The RC5 codes:

Address	Equipment
0	TV1
1	TV2
2	Videotext
3	Expansion for TV1 and TV2
4	Laser Vision Player
5	Video recorder1 (VCR1)
6	Video recorder2 (VCR2)
7	Reserved
8	SAT1
9	Expansion for VCR1 and VCR2
10	SAT2
11	Reserved
12	CD Video
13	Reserved
14	CD Photo
15	Reserved
16	Audio preamplifier1
17	Tuner
18	Analogue cassette recorder
19	Audio preamplifier2
20	CD
21	Audio Rack or Aufnahmegerät
22	Audio Satellite receiver
23	DCC Recorder
24	Reserved
25	Reserved
26	writable CD
27...31	Reserved

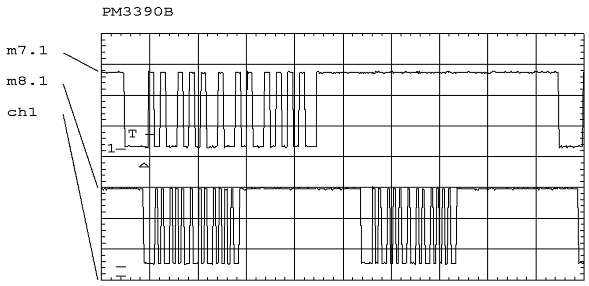
keycodes:

Code	Key Function
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
16	Volume +
17	Volume –
18	Brightness +
19	Brightness –
20	Colour saturation +
21	Colour saturation –
22	Bass +
23	Bass –
24	Treble +
25	Treble –
26	Balance right
27	Balance left
63	System select
71	Dim local display
77	Linear function increment
78	Linear function decrement
80	Step up
81	Step down
82	Menu on
83	Menu off
84	Display A/V system status
85	Step left
86	Step right
87	Acknowledge
88	PIP on/off (Pay TV channel + for system 3)
89	PIP shift (Pay TV channel - for system 3)
90	PIP / main swap (Radio channel + for system 3)
91	Strobe on/off (Radio system – for channel 3)
92	Multi strobe (Date + for system 9)
93	Main frozen (Date – for system 9)
94	3/9 multi-scan (Start time + for system 9)
95	PIP select (Start time – for system 9)
96	Mosaic/multi-PIP (Record program + for system 9)
97	Picture DNR (Record program – for system 9)
98	Main stored (Alternate channel for system 9)
99	PIP strobe (Stop time + for system 9)
100	Recall main picture (Stop time – for system 9)
101	PIP freeze
102	PIP step up +
103	PIP step down –
118	Sub mode
119	Options sub mode
123	Connect
124	Disconnect

Special commands for equipment addresses 0 und 1 (TV1 / TV2):

Code	Key Function
10	1/2/3 digits / 10
11	Freq./prog./ch./11
12	Standby
13	Mute/de-mute
14	Personal pref.
15	Display
28	Contrast +
29	Contrast –
30	Search +
31	Tint/hue –
32	Ch./prog. +
33	Ch./prog. –
34	Altern./ch.
35	? language
36	Spatial stereo
37	Stereo/mono
38	Sleep timer
39	Tint/hue. +
40	RF switch
41	Store/execute/vote
42	Time
43	Scan fwd./increm.
44	Decrement
46	Sec con/menu
47	Show clock
48	Pause
49	Erase/correct
50	Rewind
51	Go to
52	Wind
53	Play
54	Stop
55	Record
56	External 1
57	External 2
59	Advance
60	TXT sub-mode/12
61	Sys. Standby
62	Crispener
70	Speech/music
79	Sound scroll
104	PIP size
105	Pic. Scroll
106	Act. On/off
107	Red
108	Green
109	Yellow
110	Cyan
111	Index/white
112	Next
113	Previous
122	Store open/close
126	Movie expand
127	Parental access

SIRCS/Control S Code



```
Y/Div:   Timebase:   TRACE   Trigger time & date
2.00 V    10.0ms     ch1     12:36:23:88  18-10-2000
2.00 V    5.00ms     m7.1    12:31:08:17  18-10-2000 =COPY(ch1)
2.00 V    10.0ms     m8.1    12:33:33:16  18-10-2000 =COPY(ch1)

Time of hardcopy:               12:40:54   18-10-2000
```

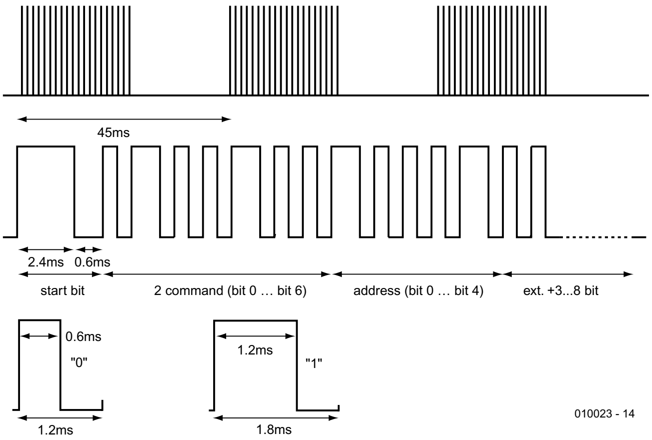
010023 - 13

Figure 3. SIRCS code at the output of the TFMS5360 receiver IC.

A message sent using the SIRCS or CNTRL S protocol from Sony consists of twelve to twenty bits. Five to thirteen of these bits is used for the address field and seven bits for the key code.

A Start bit (2.4 ms) is sent followed by a 0.6 ms space or pause. Next comes the data. A ’1’ is represented by a 1.2 ms ON or mark followed by a 0.6 ms OFF or pause. A ’0’ is represented by a 0.6 ms ON and a 0.6 ms OFF. A typical message is shown in **Figure 4**. The message is sent a minimum of twice (five times for a camcorder). The message is discarded if an error is detected. SIRCS message coding is identical to CNTRL S, but SIRCS modulates the code at 40 kHz ready to be sent to an IR diode. CNTRL-S is the baseband (unmodulated) signal and is used between equipment where a communications cable is fitted. Sony produce the following IC:

Transmitter:
KIE RA275 S42



010023 - 14

Figure 4. CNTRL-S and SIRCS message format.

The Sony codes:

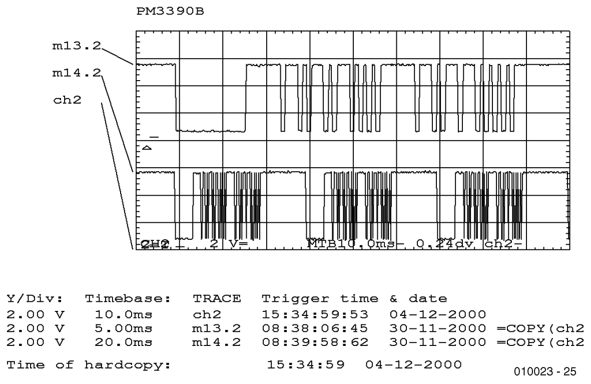
Equipment address codes (decimal):

Address	Equipment type
1	TV
2	VTR1
4	VTR2
6	Laserdisc
7	VTR2
11	VTR3
12	Surround sound processor
16	Cassette deck, tuner
17	CD Player
18	Equaliser
164	TV digital effects (8 bit device code)

keycodes:

code	key Function
000	1 button
001	2 button
002	3 button
003	4 button
004	5 button
005	6 button
006	7 button
007	8 button
008	9 button
009	10 button/0 button
011	Enter
016	Channel up
017	Channel down
018	Volume up
019	Volume down
020	Mute
021	Power
022	Reset TV
023	Audio mode: mono/SAP/stereo
024	Picture up
025	Picture down
026	Colour up
027	Colour down
030	Brightness up
031	Brightness down
032	Hue up
033	Hue down
034	Sharpness up
035	Sharpness down
036	Select TV tuner
038	Balance left
039	Balance right
041	Surround on/off
042	Aux/Ant
047	Power off
048	Time display
054	Sleep timer
058	Channel display
059	Channel jump
064	Select input video1
065	Select input video2
066	Select input video3
074	Noise reduction on/off
078	Cable/broadcast
079	Notch filter on/off
088	PIP channel up
089	PIP channel down
091	PIP on
092	Freeze screen
094	PIP position
095	PIP swap
096	Guide
097	Video setup
098	Audio setup
099	Exit setup
107	Auto program
112	Treble up
113	Treble down
114	Bass up
115	Bass down
116	+ key
117	– key
120	Add channel
121	Delete channel
125	Trinitone on/off
127	Displays a red RtestS on the screen

RECS80 Code



```
Y/Div:   Timebase:   TRACE   Trigger time & date
2.00 V   10.0ms      ch2     15:34:59:53  04-12-2000
2.00 V   5.00ms      m13.2   08:38:06:45  30-11-2000  =COPY(ch2
2.00 V   20.0ms      m14.2   08:39:58:62  30-11-2000  =COPY(ch2
Time of hardcopy:            15:34:59   04-12-2000
```

Figure 5. RECS80 code at the output of the TFMS5360 receiver IC.

The RECS80 code from Philips a pulse position modulation technique. With this system a fixed length pulse of light is followed by a variable length space. The space timing conveys the data. There are 1280 possible codes divided into 64 commands and 20 subsystems. A subsystem is simply the type of equipment being controlled i.e. a TV or a VCR. A message is composed of 11 bits. The first two bits are the toggle bits followed by three sub-system address bits and six data bits, these indicate which key was pressed. The toggle bits are incremented if a key is released for a minimum time but will remain unchanged within a multiple key-stroke sequence. If the transmitter is configured to operate in modulated mode the first toggle bit is replaced by a REF bit of fixed duration.

In the lower trace of Figure 15 not all the data is shown because of the low sampling rate of the scope some of the bits have been missed. The RECS80 protocol encodes the data by variable length spaces between constant width ON pulses (140.8 µs). If the transmitter is configured to modulation mode this time period will be represented by a burst of carrier frequency. If configured to flash mode the IR transmitter will be flashed on at this time. A ‘0’ has a space of 5.06 ms while a ‘1’ has a space of 7.60 ms (derived from a 455 kHz resonator in the remote control transmitter). Although the length of the data packet is dependent on the commands sent, the time between two messages is fixed at 121 ms. The modulation frequency is 38 kHz.

Typical ICs for remote control:

Transmitter:
SAA3004, SAA3007 and SAA3008 (Philips)
M3004, M3005, M3006 (ST Microelectronics)

Receiver:
SAA3009, SAA3049 (Philips)

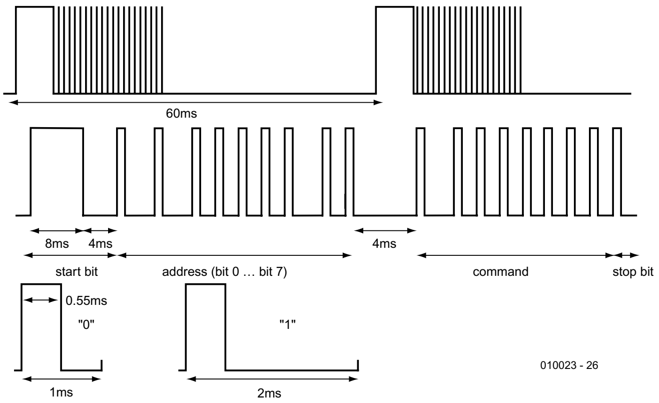


Figure 6. RECS80 code message format

Next month:

– NEC
– DENON
– MOTOROLA
– JAPAN
– SAMSUNG
– DAEWOO

Internet Links

Nec Format:
http://www.princeton.com.tw/spechtml/remote/2221.htm

Philips semiconductor:
http://www.semiconductors.com
www-us.semiconductors.com/pip/SAA3049AP

Motorola Home-Page:
http://motorola.com
Motorola-Format:
http://holtek.com

Samsung Home-Page:

http://www.intl.samsungsemi.com/System_LSI/Microcontroller/Product_Guide/Microcontroller/product_guide.html

Running Text Display with Seven LEDs

By F. Wohlrabe

In the search for original running text displays that catch your eye, manufacturers keep coming up with interesting variations that combine maximum functionality with minimum cost.

Conventional running text displays use various sizes of arrays of light-emitting diodes or other optical display elements. For example, the Running Text Display described in the February 2000 Supplement of *Elektor Electronics* contains a total of 245 LEDs (7 × 35) in order to be able to display up to six characters at the same time. In order to keep current consumption, wiring effort and component costs within limits, the light-emitting diodes are operated in multiplexed mode in nearly all running text displays. This means that individual rows, consisting of seven diodes arranged in a vertical line, are driven at a rate of 1000 Hz (for example) and thus are lit for only around 1 ms.

In **Figure 1**, you can see that power is applied to the first row at time t_1 and the information to be displayed is output at the same time (two LEDs on). After one millisecond, row 1 is switched off and row 2 is driven with the information that it has to display. The desired information is complete after 5 ms. Although only seven LEDs at most are driven at the same time, the visual inertia of the eye causes you to see an image composed of many more points of light.

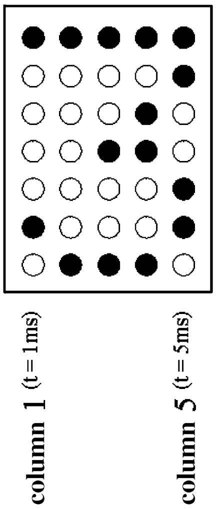


Figure 1. Driving the rows of a running text display.

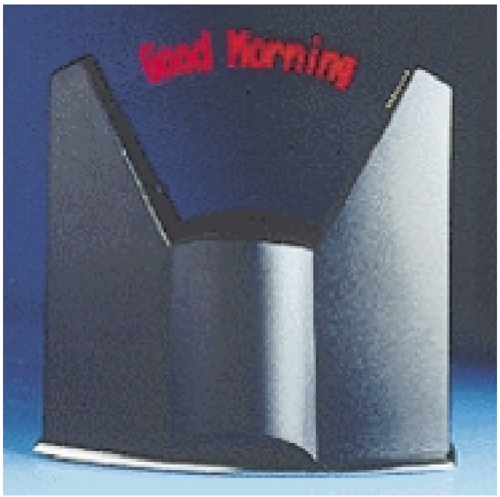


Figure 2. ‘Free-floating’ running text.

In spite of multiplexed operation, it must be possible to drive each row separately, which requires a not inconsiderable amount of driver ICs or power gates. However, things can be made even simpler, as far as the electronics are concerned, if we want to have running text that appears to float freely in the air. The inertia of the eye is also exploited here to combine many individual images into a single pattern. The image shown above (‘Good morning’) is built up using only seven LEDs. The trick here is to repeatedly displace these diodes by a small amount in order to display the desired information. In this case, electronic components of the previously mentioned running text display are partly replaced by mechanical components. A circuit board that carries the LEDs arranged in a vertical line is moved quickly back and forth by a motor. The total displacement is divided into small segments using software, with specific data to be displayed by the LEDs being assigned to each segment. The start of a sweep can for example be optically signalled to the microcontroller by a light barrier detector, or magnetically signalled by Hall sensors.

Figure 3 shows how the numeral ‘3’ is displayed using seven LEDs. The starting signal for the microcontroller to output a pattern is the switch closure, and LEDs 2 and 7 (counting from the bottom) are driven at this time. Every following millisecond, when the circuit board has moved a few millimetres, a new pattern is sent out.

The amount of information that can be displayed depends on the length of the circuit board and how far along the circuit board the LEDs are located. The mechanical robustness of such a system is naturally a good indication of its life expectancy, and thus its quality. In order to produce a clean, stationary image that does not appear 'washed out', the display board must be moved at a high speed so that only the LEDs are visible, and not the circuit board.

There are even 'breakdown help batons' available for car drivers. These are fitted with seven LEDs and are supposed to be waved manually back and forth in front of the body. When they are moved this way, they generate the message 'Help', by means of which the hapless driver of a broken down vehicle can call attention to himself or herself (note that is may not be allowed in all countries in which *Elektor Electronics* is published — *Ed*.)

To find out more about what can be displayed using only seven LEDs, look at the home page of Bob Blick at http://bobblick.com/bob/projects/. It explains the mechanical and electronic construction of clocks and running text displays that operate on the principle described here.

(000194-1)

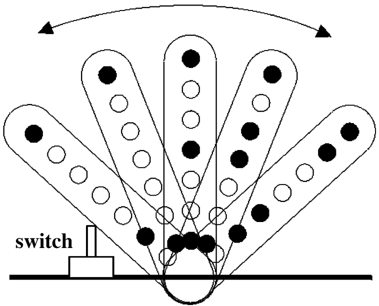


Figure 3. Basic motion of the carrier circuit board with seven LEDs.

Desoldering Tip

By Charles Verschuur

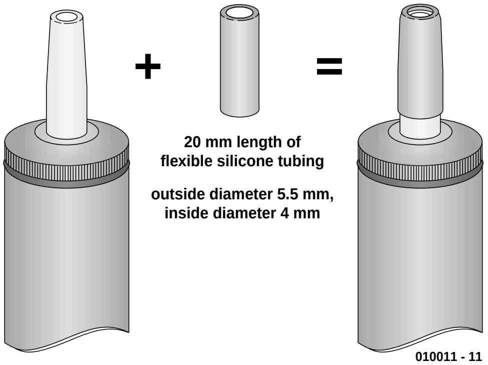
icharles@iafrica.com

Some desoldering tasks are just downright stubborn. Although you apply more than enough heat, that last bit of solder will just not budge. Sometimes the PCB tracks and connections are so delicate that disaster is imminent when attempting to use a ‘normal’ hard-tipped handheld solder sucker. With the simple modification shown here you stand to experience much less frustration at the repair, maintenance or prototyping bench.

Simply wipe the solder sucker tip clean, slide a 15-20 mm length of flexible silicone rubber (or Teflon®) tubing over the tip so that approximately 3 mm extends beyond the original tip. You then have a super solder sucker with a soft tip, which will become invaluable as a user and circuit-friendly tool.

Silicone tubing may be obtained in various diameters from modellers shops. In petrol-powered models, silicone tubing is often used between the petrol tank and the carburettor. Teflon® tubing may be harder to find, look around in the amateur radio trade where it is used for high-voltage and SHF circuits and antenna finishing.

(010011-1)

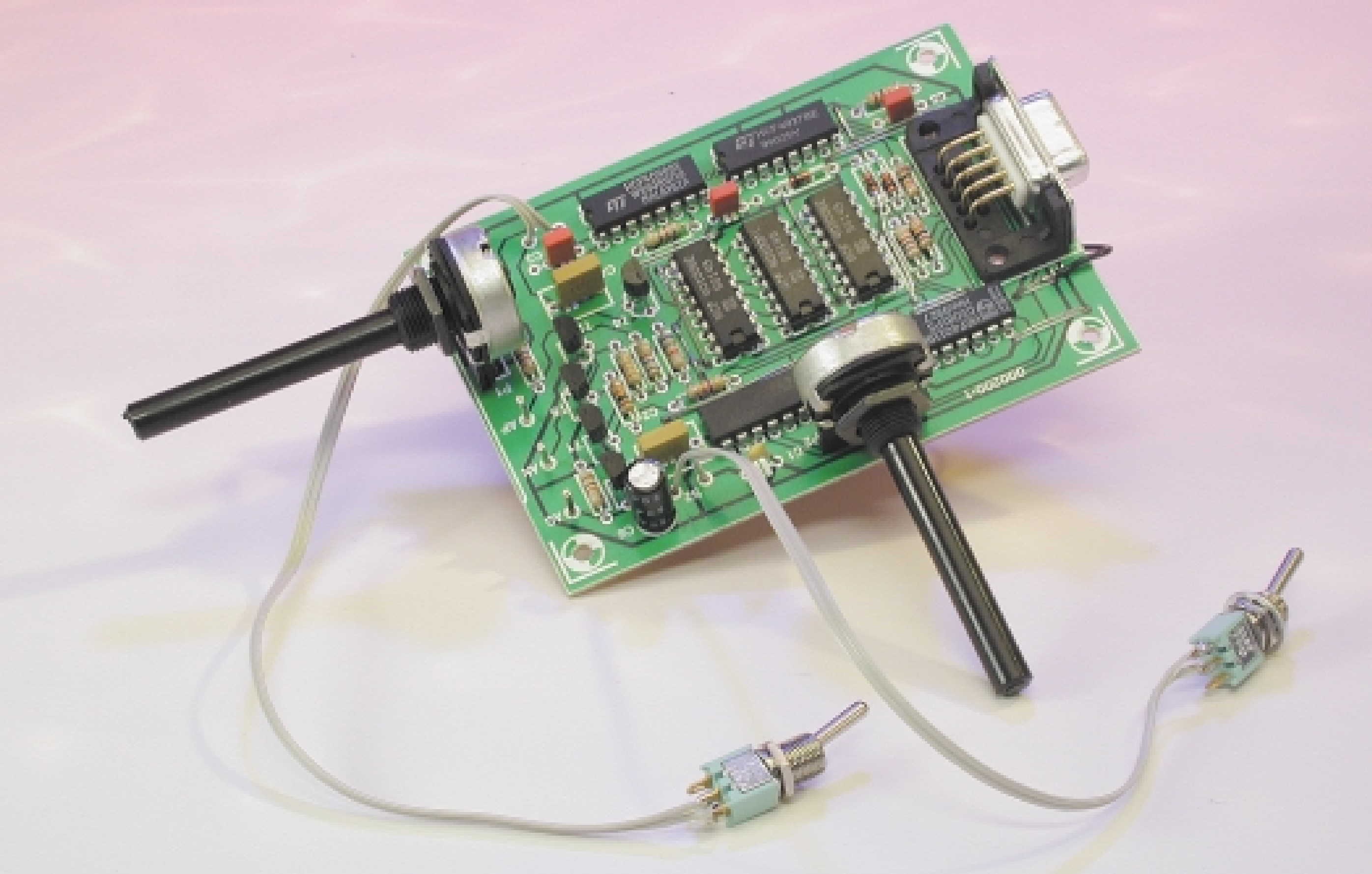


Programmable Pulse Generator

simple hardware with lots of features

Design by B. Oehlerking

This pulse generator, which can be programmed using Visual Basic 5.0, employs the simplest imaginable hardware but offers capabilities that are usually found only in microprocessor controlled equipment.



The circuit presented here is a programmable pulse generator (PPG) that can be programmed via a serial interface. It has a frequency range of 0.1 Hz to around 100 kHz. The pulse frequency and pulse length of the PPG are set using the combination of switch S2 (SPF) and potentiometer P2 (PF) along with switch S1 (SPL) and potentiometer P1 (PTL), as shown in **Figure 1**. It is possible to program arbitrary pulse sequences, ranging in length from 1

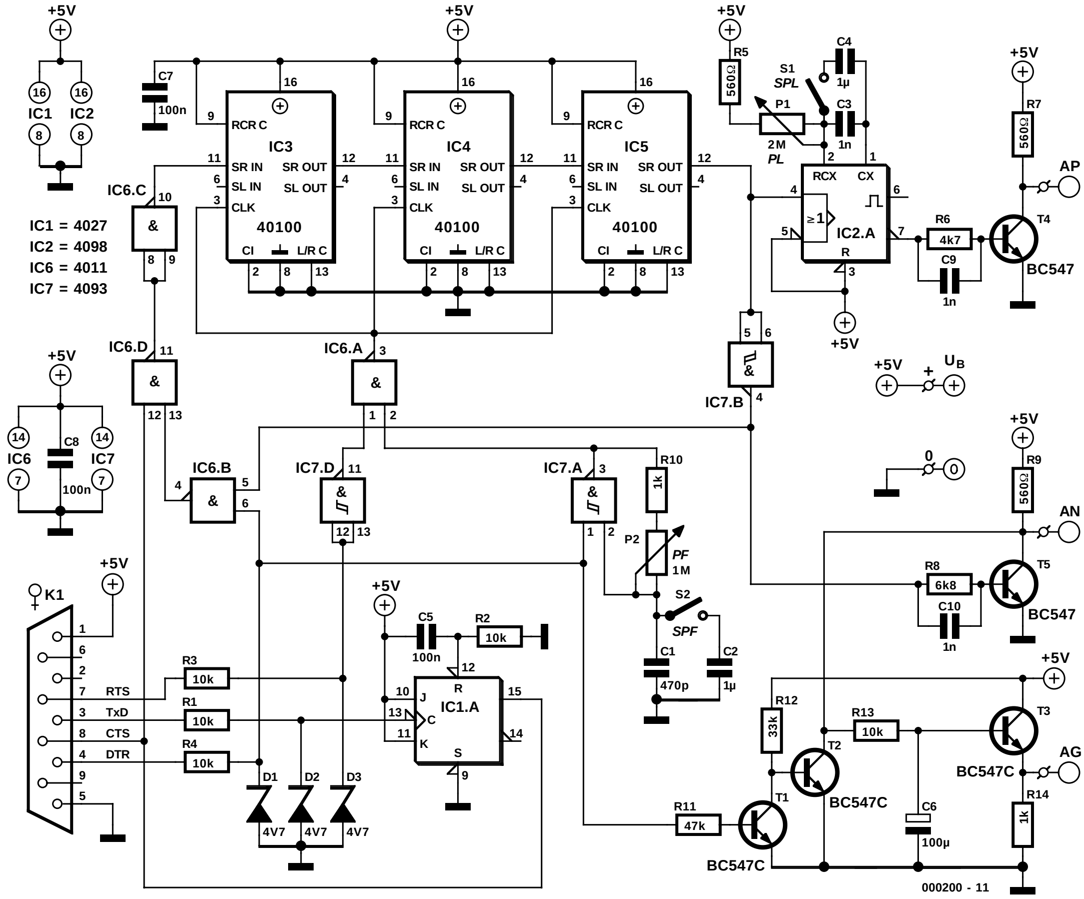


Figure 1. Only a few standard logic ICs are needed for the hardware of the programmable pulse generator.

to 48 pulses, and the only involvement of the PC CPU is to load new pulse patterns. The circuit is otherwise fully independent.

The following signals are present on the three outputs (AN, AG and AP): the programmed pulse sequence, a DC voltage smoothed by a low-pass filter (dimming function) and a pulse whose width can be set using a monostable multivibrator (MMV).

The software, which has been generated using VB 5.0, consists of a BAS module and three Form modules. Two of the Form modules represent the pulse generator and a user interface for testing, while the third Form module indicates whether the interface connector is properly connected and the correct operating voltage is present. The BAS module contains procedures and functions that are useful for putting together a graphical user interface. The processing speed of the PC affects only the loading of a new pulse sequence, so it does not matter as far as the actual operation of the circuit is concerned.

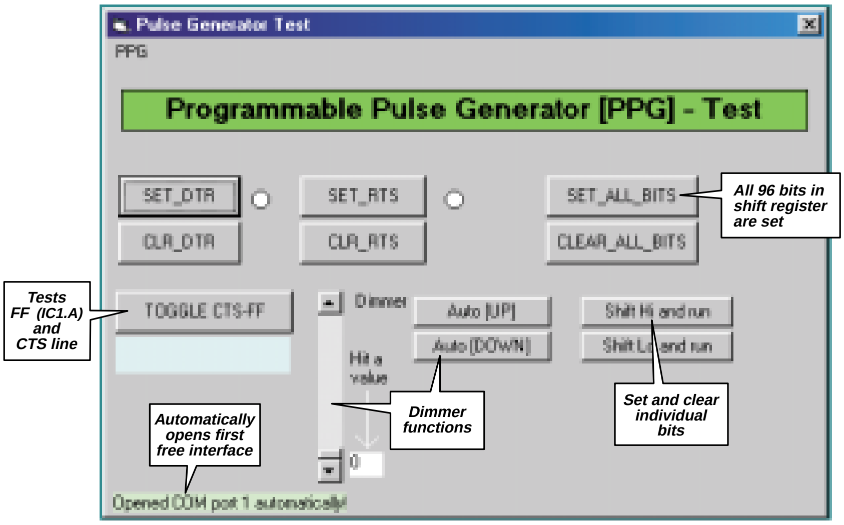
Operating principle

The software loads a specific pulse pattern into three 32-bit shift register ICs (IC3–IC5, type 40100). After the pattern has been loaded, the software enables a clock generator, and the pulse sequence recirculates through the shift registers, whose output is coupled back to the input.

To better understand how this works, consider an example in which a single pulse is loaded and then recirculates through the shift register chain. Once the PC and +5-V supply voltage have been switched on and the serial interface cable has been connected, the DTR signal lies at a low voltage, which disables the oscillator circuit formed by IC7a, P2, R10, C1 and C2. Pin 2 of gate IC6a is then High. Since the RTS line also supplies a Low level, the common clock line for the shift register (IC3–IC5) is also held Low (pin 3). The software now sends pulses via TxD until pin 15 of IC1a is set. This is recognised by the software from the state of the CTS line. The High level at pin 15 of IC1a reaches the input of the shift register chain (SR_IN) via gate IC6d and inverter IC6c. The software now sets RTS briefly High and then switches it back to Low. This produces a complete pulse on the clock line of the shift register, so that the High level at its input is transferred to the first stage. Following this, pin 15 of IC1a must be set Low by a suitable signal on the TxD line. A series of 95 zeroes is then clocked into the shift register by 95 clock pulses supplied via the RTS line. This means that 95 stages of the shift register are Low, with the final stage being High.

Up to now, the DTR signal has remained Low and thus held the voltage at the AN and AG outputs at zero (via T1 and T2). Pin 15 of IC1a is now set High via the TxD line, and then the clock generator is enabled by setting DTR High. This allows a single pulse to recirculate through the shift register chain via gates IC7b, IC6b, IC6d and IC6c. This can all be clearly represented using a table or flow chart, as shown in the text box.

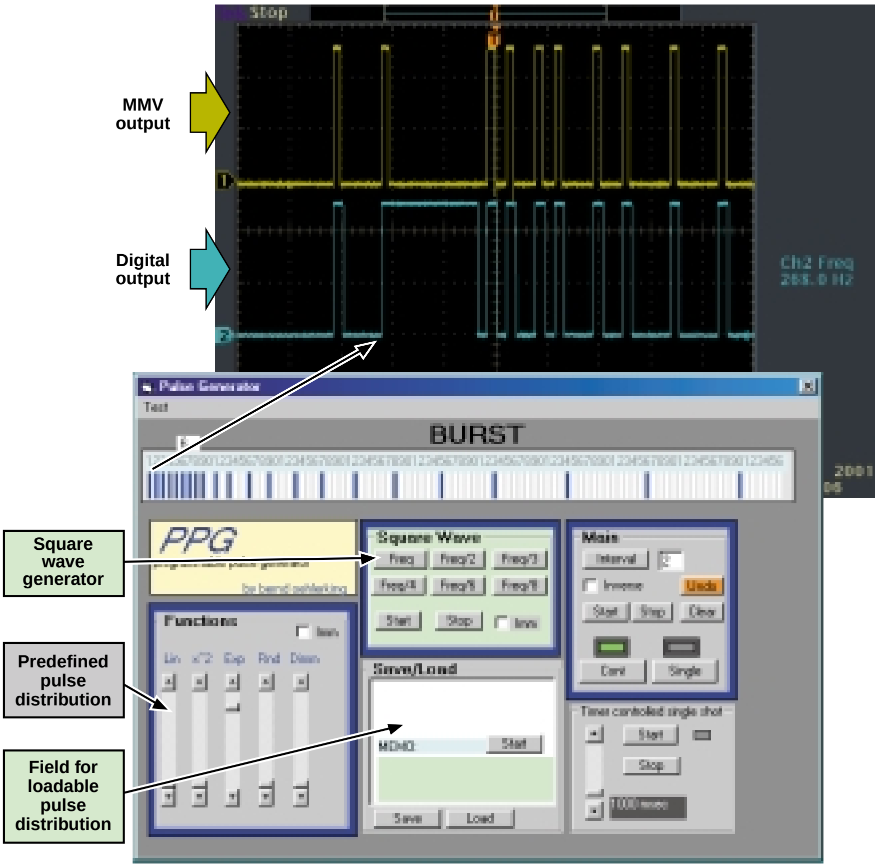
Since the voltages provided by PC serial interfaces are usually greater than +5 V or negative, all of the inputs to the circuit (TxD, DTR


Pulse Generator Test
PPG
Programmable Pulse Generator [PPG] - Test
SET_DTR
CLR_DTR
SET_RTS
CLR_RTS
SET_ALL_BITS
CLEAR_ALL_BITS
All 96 bits in shift register are set
Tests FF (IC1.A) and CTS line
TOGGLE CTS-FF
Dimmer
Auto [UP]
Auto [DOWN]
Hit a value
0
Shift Hi and run
Shift Lo and run
Dimmer functions
Set and clear individual bits
Automatically opens first free interface
Opened COM port 1 automatically!

000200- 12

Figure 2. The user interface of the test program, with comments.

and RTS) are protected by resistors and Zener diodes. The DCD line (pin 1) is used to show the user whether the interface is connected and the power supply is switched on. The upper frequency limit of the circuit is determined by the maximum clock rate of the 40100 shift register IC, which is specified to be 1 MHz (at +5 V). With other component types, it might be possible to improve the performance in this regard.



000200 - 13

Figure 3. The main window for the programmable pulse generator, with the associated oscillogram representation.

Graphical user interface

The software modules that have already been mentioned include a user interface for testing the finished circuit (see **Figure 2**). This text window should be self-explanatory.

The main window of the PPG is shown in **Figure 3**. This displays a configured pulse pattern in the Burst window, above which the resulting oscillogram can be seen. The upper channel shows the pulse pattern, while the lower channel shows a monostable pulse sequence that has been configured at the AP output of the circuit using potentiometer P2. Pulse patterns can be simply selected from the Burst window using the mouse and then loaded into the hardware with ‘Start’. A complete group of pulses can be selected by clicking on the group with the left mouse button, dragging the mouse to the right and then clicking with the right mouse button. Such a group can subsequently be deleted by clicking on a pulse in the group, moving the mouse to the left and then clicking with the right mouse button.

Board layout

Such an interesting piece of test and measurement equipment certainly deserves its own circuit board layout. As can be seen in **Figure 4**, this is an uncomplicated design with plenty of space between the tracks and the individual components. Since the board is single sided, a few (well, ten) wire bridges are unavoidable. You should start board assembly with these items, followed by the low-profile components and then (good) sockets for all ICs. Finally, fit solder posts for AG, AN and AP and add the Sub-D connector and the potentiometers. Then insert the ICs and check the orientations of all polarised components. Before fitting the circuit board into an inexpensive plastic enclosure, you should solder 1-nF capacitors across R6 and R8 on the bottom side of the circuit board. After the board layout had already been completed, it was found that

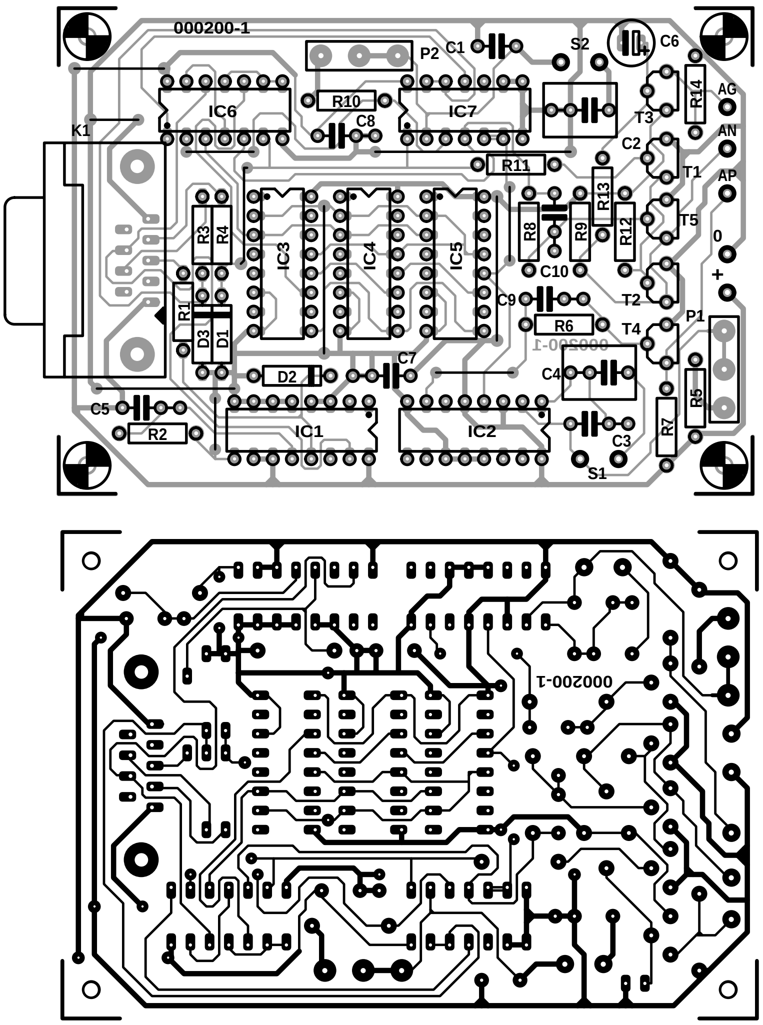


Figure 4. A printed circuit board layout for the hardware.

this measure has a beneficial effect on the performance at frequencies above 100 kHz. After completing the assembly, use an oscilloscope to check the operation of the hardware and software.

(000200-1)

COMPONENTS LIST

Resistors:

R1-R4,R13 = 10kΩ
R5,R7,R9 = 560Ω
R6 = 4kΩ7
R8 = 6kΩ8
R10,R14 = 1kΩ
R11 = 47kΩ
R12 = 33kΩ
P1 = potentiometer, 2MΩ, linear law
P2 = potentiometer, 1MΩ, linear law

Capacitors:

C1 = 470pF
C2,C4 = 1µF
C3 = 1nF
C5,C7,C8 = 100nF
C6 = 100µF 16V radial

Semiconductors:

D1,D2,D3 = 4V7, 500mW zener diode
T1-T5 = BC547C
IC1 = 4027
IC2 = 4098
IC3,IC4,C5 = 40100
IC6 = 4011
IC7 = 4093

Miscellaneous:

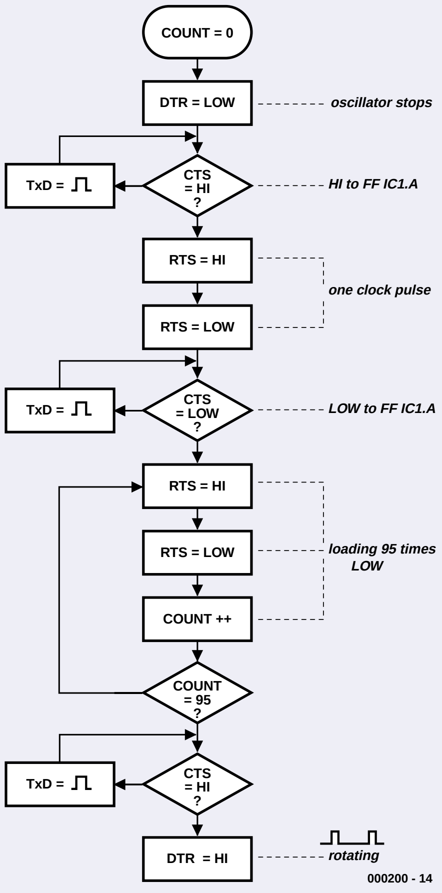
K1 = 9-way Sub-D socket (female), PCB mount
S1,S2 = switch, 1 on/off contact
3 solder pins
Disk, project software, order code **000200-11a/b** (see Readers Services page)
PCB, order code **000200-1** (see Readers Services page)

Freeware!

The program and source code for the programmable pulse generator are available free of charge from the Free Downloads page of the *Elektor Electronics* website at http://www.elektor-electronics.co.uk. Look for item **000200-11** under March 2001. You can also obtain the circuit board layout in PDF format from the website. If you do not have access to the Internet, you can obtain the software from the Publishers' Readers Services on two diskettes (order numbers **000200-11a/b**), as well as a blank circuit board (order number **000200-1**).

Programming sequence

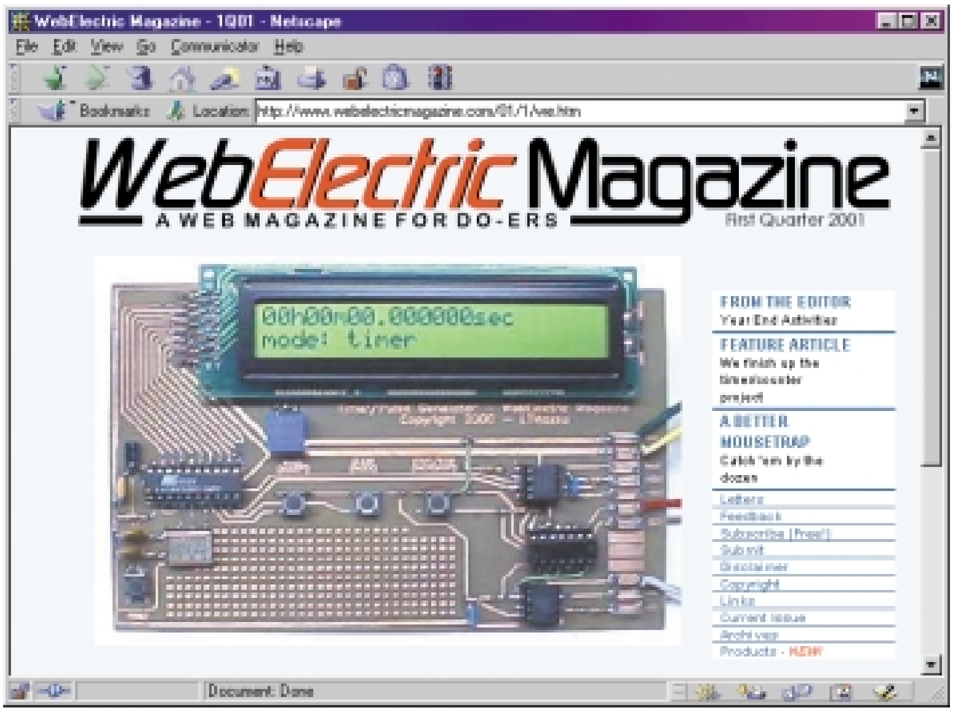
1) Disable clock generator. DTR Low
2) Load 96 states:
0 IC1a Low, RTS = High–Low clock
1 IC1a HIgh, RTS = High–Low clock
3) Enable gate IC6d IC1a High
4) Enable IC6b and clock generator.
DTR High



Circuits for home construction

Circuit diagrams galore

One great advantage of the Internet is rapid availability of information you may need at any moment. If you want to build an electronic circuit for a specific application, you're likely to find a suitable schematic on the Internet. And it's especially easy if you bookmark the websites mentioned in this article.



In the May 1998 instalment of 'Electronics Online' we discussed electronics circuit collections on the Internet. However, that is almost three years ago, a very long time in Internet terms. The subject remains popular, however, so we thought that an update was due.

Imageering on-line magazine [1] is an electronic magazine (e-zine) run by electronics engineer David Johnson. Apart from masses of electronics-related stuff, this website also has a 'hobby' corner neatly divided into hobby schematics, ideas, suggestions and tips on electronics. David is a devoted home brewer (of electronics, of course) and has designed many circuits during his 30-year career. A number of his circuits have made it to publication in the electronics press. The hobby corner contains a large number of circuit diagrams, while the ideas section contains more than 100 descriptions of circuits that have not been (completely) realised. Anyone is free to contribute to the best of his/her ability!

Web-Electric Magazine [2] is an electronics webzine totally dedicated to home brewers. Each quarter, a new circuit appears on this website and is discussed in detail. As we write this, the subject is a timer/pulse generator. This website also contains an archive from which circuits published over the last three years may be retrieved. The circuit descriptions are extensive and neatly laid out, allowing even beginners to get started.

4QD [3] is a company involved in the design and production of speed controllers for small electromotors. Their website hosts an electronics 'club' (4QD-TEC) which is home to a fair amount of circuits. Although the club is marked by a membership fee, there is a freely accessible archive of electronic circuits. Most circuits found here are small, but then again that may be just what you were after in the first place.

Tony's Website [4] published by Tony van Roon of Guelph University, Ontario, Canada, also contains two pages of circuits for DIY construction. The range is from simple flashing circuits to (illegal) FM transmitters.

The Circuit Archive [5] at the University of Washington is still highly recommended. The available schematics are neatly arranged by subject. The quality of the schematics differs widely, mainly because there are many links taking you to other

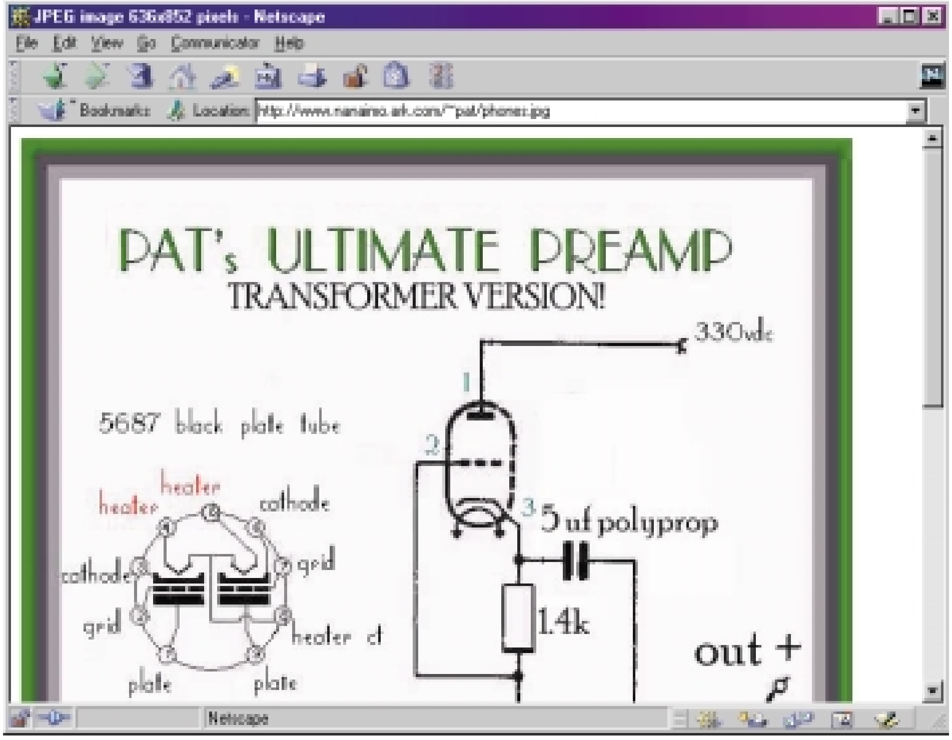
Internet addresses. Besides those circuit diagrams, this website also contains many links to component datasheets and other electronics-related stuff.

Jack Orman's **Digital Music Zone [6]** contains circuit diagrams of music-related electronics ('electrophonics') including a distortion unit, a wah-wah filter, a phaser and a flanger. We also found circuits to build analogue and digital delays.

Pat's Tube & Recording Schematics [7] is an interesting address for everyone involved in the design and construction of valve (tube) amplifiers. Although Pat claims that more than 4000 schematics may be found on his site, that number should taken to mean including all hyperlinks to other websites. Nonetheless, Pat's pages contain a number of nice designs, including a headphone amplifier and a microphone preamplifier.

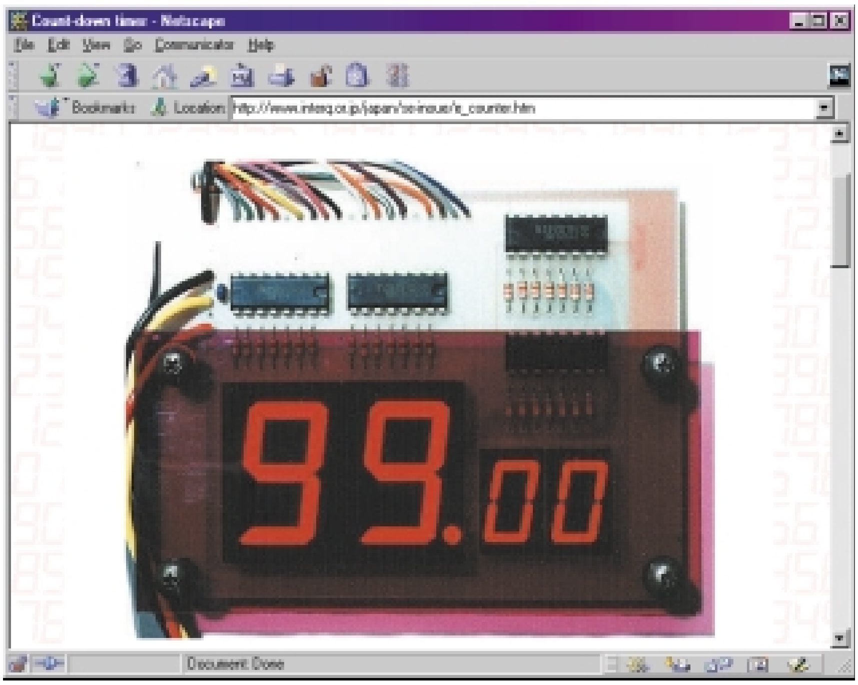
Aaron Cake [8] is another fervent electronics enthusiast. In addition to spending time on his hobby, Aaron also puts a lot of time and effort in the general design of his website. We found a rather long list of circuit diagrams. Each of these is complete with a parts list, a brief description and, in many cases, a PCB layout.

The Electronic Circuits Application Garage [9] is a webpage run by Seiichi Inoue from Japan. Besides a number of electronics-related subjects (including a datasheet collection and a PIC design guide) the Garage contains some fully worked out projects like an ultrasonic distance meter and a 4-channel oscilloscope.



We could go on listing more 'circuit' websites, but ten of these ought to be enough this month! Number 10, then, is for **Jordan's RF Web Site [10]**. As indicated by the title, this is mostly on radio-related (RF) designs. Some 25 designs are available including dividers, oscillators, RF amplifiers and measurement instruments. Jordan's main project is an RF tracking generator from 100 kHz to 1 GHz. This instrument is described in great detail. Now, if this month's *Elektor Electronics* did not give you enough inspiration, you know you can start looking 'elsewhere'.

(015029-1)


Count-down timer - Netscape
File Edit View Go Communicator Help
Bookmarks
Location: http://www.interq.or.jp/japan/se-inoue/e_counter.htm
99.00
Document: Done

[1] Imagineering on-line magazine:
www.imagineeringezine.com/

[2] Web-Electric Magazine:
www.webelectricmagazine.com/01/1/we.htm

[3] 4QD-TEC:
www.4qd.co.uk/ccts/index.html

[4] Tony's website:
www.uoguelph.ca/~antoon/circ/circuits.htm

[5] Circuits Archive at Washington University:
www.ee.washington.edu/circuit_archive/circuits/

[6] Digital Music Zone:
http://members.aol.com/jorman/schem.html

[7] Pat's Tube & Recording Schematics:
www.nanaimo.ark.com/~pat/index.htm

[8] Circuits Page, Aaron Cake:
www.aaroncake.net/electronics/index.html

[9] Electronic Circuits Application Garage:
www.interq.or.jp/japan/se-inoue/e_cktsmpl.htm

[10] Jordan's RF Web Site:
www.geocities.com/jordan30bg/index.html